XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos
24 a 28 de maio de 2010
Gramado, RS

Minicursos
Livro Texto

Editora
Sociedade Brasileira de Computação (SBC)

Organizadores
Carlos Alberto Kamienski (UFABC)
Luciano Paschoal Gaspary (UFRGS)
Marinho Pilla Barcellos (UFRGS)

Realização
Instituto de Informática
Universidade Federal do Rio Grande do Sul (UFRGS)

Promoção
Sociedade Brasileira de Computação (SBC)
Laboratório Nacional de Redes de Computadores (LARC)

Capa: Josué Klafke Sperb
Produção Editorial: Flávio Roberto Santos, Roben Castagna Lunardi, Matheus Lehmann, Rafael Santos Bezerra, Luciano Paschoal Gaspary e Marinho Pilla Barcellos.

Cópias Adicionais:
Sociedade Brasileira de Computação (SBC)
Av. Bento Gonçalves, 9500 - Setor 4 - Prédio 43.412 - Sala 219
Bairro Agronomia - CEP 91.509-900 - Porto Alegre - RS
Fone: (51) 3308-6835
E-mail: sbc@sbc.org.br

Dados Internacionais de Catalogação na Publicação (CIP)

Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (28. : 2010 : Gramado, RS).

Minicursos / XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos ; organizadores Carlos Alberto Kamienski, Luciano Paschoal Gaspary, Marinho Pilla Barcellos. − Porto Alegre : SBC, c2010.
240 p.

ISSN 2177-4978

1. Redes de computadores. 2. Sistemas distribuídos. I. Kamienski, Carlos Alberto. II. Gaspary, Luciano Paschoal. III. Barcellos, Marinho Pilla. IV. Título.

Promoção

Sociedade Brasileira de Computação (SBC)

Diretoria

Presidente
José Carlos Maldonado (USP)

Vice-Presidente
Marcelo Walter (UFRGS)

Diretor Administrativo
Luciano Paschoal Gaspary (UFRGS)

Diretor de Finanças
Paulo Cesar Masiero (USP)

Diretor de Eventos e Comissões Especiais
Lisandro Zambenedetti Granville (UFRGS)

Diretora de Educação
Mirella Moura Moro (UFMG)

Diretora de Publicações
Karin Breitman (PUC-Rio)

Diretora de Planejamento e Programas Especiais
Ana Carolina Salgado (UFPE)

Diretora de Secretarias Regionais
Thais Vasconcelos Batista (UFRN)

Diretor de Divulgação e Marketing
Altigran Soares da Silva (UFAM)

Diretor de Regulamentação da Profissão
Ricardo de Oliveira Anido (UNICAMP)

Diretor de Eventos Especiais
Carlos Eduardo Ferreira (USP)

Diretor de Cooperação com Sociedades Científicas
Marcelo Walter (UFRGS)

Realização

Comitê de Organização

Coordenação Geral
Luciano Paschoal Gaspary (UFRGS)
Marinho Pilla Barcellos (UFRGS)

Coordenação do Comitê de Programa
Luci Pirmez (UFRJ)
Thaís Vasconcelos Batista (UFRN)

Coordenação de Palestras e Tutoriais
Lisandro Zambenedetti Granville (UFRGS)

Coordenação de Painéis e Debates
José Marcos Silva Nogueira (UFMG)

Coordenação de Minicursos
Carlos Alberto Kamienski (UFABC)

Coordenação de Workshops
Antônio Jorge Gomes Abelém (UFPA)

Coordenação do Salão de Ferramentas
Nazareno Andrade (UFCG)

Comitê Consultivo
Artur Ziviani (LNCC)
Carlos André Guimarães Ferraz (UFPE)
Célio Vinicius Neves de Albuquerque (UFF)
Francisco Vilar Brasileiro (UFCG)
Lisandro Zambenedetti Granville (UFRGS)
Luís Henrique Maciel Kosmalski Costa (UFRJ)
Marcelo Gonçalves Rubinstein (UERJ)
Nelson Luis Saldanha da Fonseca (UNICAMP)
Paulo André da Silva Gonçalves (UFPE)

Organização Local
Adler Hoff Schmidt (UFRGS)
Alan Mezzomo (UFRGS)
Alessandro Huber dos Santos (UFRGS)
Bruno Lopes Dalmazo (UFRGS)
Carlos Alberto da Silveira Junior (UFRGS)
Carlos Raniery Paula dos Santos (UFRGS)
Cristiano Bonato Both (UFRGS)
Flávio Roberto Santos (UFRGS)
Jair Santanna (UFRGS)
Jéferson Campos Nobre (UFRGS)
Juliano Wickboldt (UFRGS)
Leonardo Richter Bays (UFRGS)
Lourdes Tassinari (UFRGS)
Luís Armando Bianchin (UFRGS)
Luis Otávio Luz Soares (UFRGS)
Marcos Ennes Barreto (UFRGS)
Matheus Brenner Lehmann (UFRGS)
Pedro Arthur Pinheiro Rosa Duarte (UFRGS)
Pietro Biasuz (UFRGS)
Rafael Pereira Esteves (UFRGS)
Rafael Kunst (UFRGS)
Rafael Santos Bezerra (UFRGS)
Ricardo Luis dos Santos (UFRGS)
Roben Castagna Lunardi (UFRGS)
Rodolfo Stoffel Antunes (UFRGS)
Rodrigo Mansilha (UFRGS)
Weverton Luis da Costa Cordeiro (UFRGS)

Mensagem dos Coordenadores Gerais

Bem-vindo(a) ao XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2010)! Esta edição do simpósio está sendo realizada de 24 a 28 de maio de 2010 na pitoresca cidade de Gramado, RS. Promovido pela Sociedade Brasileira de Computação (SBC) e pelo Laboratório Nacional de Redes de Computadores (LARC) desde 1983, o SBRC 2010 almeja não menos que honrar com uma tradição de quase 30 anos: ser reconhecido como o mais importante evento científico em redes de computadores e sistemas distribuídos do país, e um dos mais concorridos em Informática. Mais do que isso, pretende estimular intercâmbio de idéias e discussões qualificadas, aproximá-lo(a) de temas de pesquisa efervescentes e fomentar saudável aproximação entre estudantes, pesquisadores, professores e profissionais.

Para atingir os objetivos supracitados, reunimos um grupo muito especial de professores atuantes em nossa comunidade que, com o nosso apoio, executou com êxito a tarefa de construir um **Programa Técnico** de altíssima qualidade. O SBRC 2010 abrange as seguintes atividades: 20 sessões técnicas de artigos completos, cobrindo uma grande gama de problemas em redes de computadores e sistemas distribuídos; 2 sessões técnicas para apresentações de ferramentas; 5 minicursos ministrados de forma didática, por professores da área, sobre temas atuais; 3 palestras e 3 tutoriais sobre tópicos de pesquisa avançados, apresentados por especialistas nacionais e estrangeiros; e 3 painéis versando sobre assuntos de relevância no momento. Completa a programação técnica a realização de 8 *workshops* satélites em temas específicos: WRNP, WGRS, WTR, WSE, WTF, WCGA, WP2P e WPEIF. Não podemos deixar de ressaltar o **Programa Social**, organizado em torno da temática "*vinho*", simbolizando uma comunidade de pesquisa madura e que, com o passar dos anos, se aprimora e refina cada vez mais.

Além da ênfase na qualidade do programa técnico e social, o SBRC 2010 ambiciona deixar, como marca registrada, seu esforço na busca por excelência organizacional. Tal tem sido perseguido há mais de dois anos e exigido muita determinação, dedicação e esforço de uma equipe afinada de organização local, composta por estudantes, técnicos administrativos e professores. O efeito desse esforço pode ser percebido em elementos simples, mas diferenciais, tais como uniformização de datas de submissão de trabalhos, portal *sempre* atualizado com as últimas informações, comunicação sistemática com potenciais participantes e pronto atendimento a qualquer dúvida. O nosso principal objetivo com essa iniciativa foi e continua sendo oferecer uma elevada *qualidade de experiência* a você, colega participante!

Gostaríamos de agradecer aos membros do Comitê de Organização Geral e Local que, por conta de seu trabalho voluntário e incansável, ajudaram a construir um evento que julgamos de ótimo nível. Gostaríamos de agradecer, também, à SBC, pelo apoio prestado ao longo das muitas etapas da organização, e aos patrocinadores, pelo incentivo à divulgação de atividades de pesquisa conduzidas no País e pela confiança depositada neste fórum. Por fim, nossos agradecimentos ao Instituto de Informática da UFRGS, por viabilizar a realização, pela quarta vez, de um evento do porte do SBRC.

Sejam bem-vindos à Serra Gaúcha para o "SBRC do Vinho"! Desejamos que desfrutem de uma semana agradável e proveitosa!

Luciano Paschoal Gaspary
Marinho Pilla Barcellos
Coordenadores Gerais do SBRC 2010

Mensagem do Coordenador de Minicursos

O Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC) é o mais importante fórum da comunidade de pesquisa e desenvolvimento em Redes de Computadores e Sistemas Distribuídos no Brasil. Entre as principais atividades técnicas do SBRC encontram-se os Minicursos, que tem como objetivo a atualização em temas normalmente não cobertos nas grades curriculares ou que despertem grande interesse entre acadêmicos e profissionais. O SBRC é uma promoção do Laboratório Nacional de Redes de Computadores (LARC) e da Sociedade Brasileira de Computação (SBC).

Os Minicursos são compostos pela apresentação presencial (4 horas de aula) e pelo livro, onde cada minicurso se torna um capítulo contendo um texto de apoio produzido pelos autores. Durante os últimos anos, tem sido observado que muitos textos de minicursos tem sido intensamente usados pela comunidade, inclusive muitos deles sendo utilizados em disciplinas de graduação e pós-graduação nas universidades brasileiras. Isso demonstra que o alcance dos minicursos do SBRC é maior do que o grupo de alunos que tiveram oportunidade de estarem presentes ao SBRC para as apresentações.

Pelos seus muitos anos de bons cursos oferecidos, os Minicursos do SBRC conquistaram grande respeito e prestígio junto à comunidade em redes e sistemas distribuídos. Uma prova disso foram as 37 propostas completas recebidas nesse ano para o SBRC 2010, das quais cinco minicursos foram selecionados. Infelizmente, como ocorre em outros eventos com alta concorrência, muitas propostas de excelente qualidade não puderam ser aceitas. No entanto, nesse processo serão beneficiado os alunos que se matricularão nos minicursos e os membros da comunidade que terão à sua disposição texto de alto nível para estudos e pesquisas avançadas.

Os minicursos selecionados para o SBRC 2010 abordam temas de grande relevância atual e que vem despertando o interesse de pesquisadores, professores, alunos e usuários nos últimos anos. Os textos completos de cada minicurso compõem esse livro que está organizado em cinco capítulos. O Capítulo 1 é intitulado “Redes Complexas na Modelagem de Redes de Computadores”, de autoria de Marcelo Almiron, Heitor Ramos, Eduardo Oliveira, João Menezes, Daniel Guidoni, Pedro Stancioli, Felipe da Cunha, André e Aquino, Raquel Mini, Alejandro Frery e Antonio Loureiro. A área de redes complexas é recente e vem sendo aplicada com sucesso para compreender, modelar e encontrar respostas a problemas importantes em aplicações como ciências sociais, ciência da informação, ciências biológicas e em áreas ligadas à tecnologia, com em redes de computadores. Dois conceitos associados com redes complexas, redes sem escala e redes de mundo pequeno, já foram encontrados em alguns estudos sobre a Internet. Características topológicas da Internet, tanto em nível de roteadores quanto de sistemas autônomos, já foram apresentados na literatura como seguindo leis de potência, que caracterizam as redes sem escala. Por outro lado, redes P2P apresentam características típicas de redes de mundo pequeno, como caminhos mínimos pequenos e alto grau de agrupamento entre os nós.

O Capítulo 2, intitulado “Interconexão de Redes na Internet do Futuro: Desafios e Soluções” tem como autores Miguel Campista, Lyno Ferraz, Igor Moraes, Marcelo Lanza, Luís Costa e Otto Duarte. Embora a Internet seja um feito tecnológico do ser humano que vem operando há mais de 40 anos e os protocolos TCP/IP estão sendo usados há quase 30 anos, ainda é de extrema importância compreender as limitações do modelo atual de interconexão da Internet e as propostas para solucioná-los. A Internet

não foi concebida para atender muitas das suas demandas atuais, como mobilidade e redes com múltiplos domicílios (*multihomed*) e essas questões tem gerado desde extensões dos protocolos atuais, até propostas que alteram completamente a sua arquitetura.

O Capítulo 3, "Novas Arquiteturas de Data Center para Cloud Computing", de Fábio Verdi, Christian Rothenberg, Rafael Pasquini e Maurício Magalhães, aborda um tema que desperta o interesse tanto da comunidade científica quanto do mercado. Conhecida como Computação em Nuvem, promete oferecer aos seus usuários um modelo de serviço onde os recursos computacionais estão localizados em algum lugar que não é o computador local do usuário nem o servidor da organização à qual ele está vinculado, mas um *datacenter* em algum lugar na Internet (ou seja, na nuvem). Os usuários atualmente podem desfrutar de serviços como o Google Docs e as empresas podem usar os sistemas corporativos da Salesforce.com e os serviços de armazenamento e processamento sob demanda oferecidos pela Amazon. Essas demandas geram novos desafios e mostram limitações das tecnologias atuais de redes para *datacenters* que precisam ser solucionadas para que as aplicações atualmente disponíveis de Computação em Nuvem continuem a evoluir e ganhar popularidade.

O Capítulo 4 é intitulado "Redes Cognitivas: Um Novo Paradigma para as Comunicações Sem Fio", de autoria de Marcelo Sousa, Rafael Lopes, Waslon Lope, Marcelo de Alencar. Redes cognitivas é uma proposta recente que visa tratar do crescente problema de complexidade nas redes de comunicação, onde o sistema continuamente se adapta para aprimorar e atualizar as suas funções, com o mínimo possível de intervenção humana. Uma rede cognitiva recebe esse nome porque é equipada com um processo (cognitivo) que percebe condições atuais, planeja, decide, atua sobre essas condições e aprende com os seus erros e acertos. A principal tecnologia das redes cognitivas são os rádios cognitivos, que permitem utilizar o espectro de maneira oportunista. No entanto, enquanto os rádios cognitivos atuam nas camadas física e de enlace de dados, as redes cognitivas atuam em todas as camadas de uma arquitetura de redes.

Finalmente, o Capítulo 5 é intitulado "Redes de Sensores Aquáticas", de autoria de Luiz Vieira, Antonio Loureiro, Antônio Fernandes e Mario Campos. Redes de sensores sem fio é uma área que obteve grande volume de pesquisas nos anos 2000, devido principalmente às suas várias aplicações, como monitoramento e vigilância. Nos últimos anos, a ideia de usar redes de sensores para ambientes subaquáticos vem recebendo grande interesse, também devido às suas potenciais aplicações e aos novos desafios apresentados. Se por um lado as redes de sensores aquáticas compartilham algumas propriedades com as suas similares terrestres, por outro lado existem diferenças significativas. Por exemplo, comunicações de rádio não funcionam bem abaixo da água e, portanto devem ser substituídas por comunicações acústicas. Além disso, enquanto que as redes de sensores terrestres são em geral estáticas, as redes aquáticas se movem devido às características próprias desses ambientes, como correntes marinhas.

Gostaríamos de agradecer aos 16 membros do Comitê de Avaliação pela dedicação à tarefa de avaliação dos artigos, algumas das quais em caráter de urgência. Agradecemos também aos coordenadores gerais do SBRC 2010, professores Luciano Paschoal Gaspary e Marinho Pilla Barcellos, ambos da UFRGS, pela confiança e apoio recebidos. Finalmente, um agradecimento especial a todos os autores que prestigiaram a iniciativa de realização dos minicursos do SBRC, submetendo propostas que refletem os resultados das suas pesquisas, estudos e projetos acadêmicos.

Carlos Alberto Kamienski
Coordenador de Minicursos do SBRC 2010

Comitê de Avaliação de Minicursos

Sumário

Capítulo 1

Redes Complexas na Modelagem de Redes de Computadores

Marcelo G. Almiron, Heitor S. Ramos, Eduardo M. Oliveira,
João G. M. de Menezes, Daniel L. Guidoni, Pedro O. Stancioli,
Felipe D. da Cunha, André L. L. de Aquino, Raquel A. F. Mini,
Alejandro C. Frery e Antonio A. F. Loureiro

Abstract

This short course aims at presenting the theory of complex networks applied to models for computer networks. This theory has been successfully applied to a wide variety of areas in which systems can be suitably represented by structures formed by interconnected elements, namely networks. Network models are present in social, information and biological sciences, and in technological areas. Our goal is to present the main concepts related to characterization measurements, complex network models, and statistical methods necessary to understand and explain the computer networks behavior.

Resumo

O presente minicurso pretende abordar a teoria de redes complexas aplicada aos modelos de redes de computadores. Essa teoria tem sido aplicada em diversas áreas do conhecimento em que os sistemas podem ser representados por estruturas formadas por elementos interconectados. Essas estruturas são conhecidas como "redes" e estão presentes nas ciências sociais, ciência da informação, ciências biológicas e em áreas ligadas à tecnologia. Nosso objetivo é apresentar conceitos de medidas de caracterização, modelos de redes complexas e métodos estatísticos necessários para o entendimento do comportamento das redes de computadores.

1.1. Introdução

Estudos recentes presentes na literatura estudam a análise estatística de propriedades de grafos de larga escala (Newman 2003), em contraste com estudos anteriores que focam em grafos pequenos e tratam de propriedades individuais dos seus vértices e arestas. Nesses estudos destacam-se a presença de grafos que representam redes com centenas a milhões e até mesmo bilhões de vértices. Esse aumento de escala leva à necessidade de criar novas

técnicas e teorias, em particular as redes complexas, para melhor entendimento desses novos cenários.

Além da escala, a descoberta de algumas propriedades como *small world* e *scale free*, que são capazes de caracterizar diversas redes reais, têm impulsionado um crescente interesse de pesquisa em diversas áreas de aplicação (Wang & Chen 2003). No contexto de redes de comunicação, em especial as redes de computadores, esse interesse pode ser observado por trabalhos relacionados à modelagem através da teoria de redes complexas em áreas como:

- roteamento na Internet (IP, Sistemas autônomos (AS), ou outros níveis);
- grafos que representam as aplicações Web, como por exemplo as ligações entre blogs;
- redes ponto a ponto (P2P), *overlays* e *exchanges*;
- correio eletrônico e outros tipos de comunicação entre os usuários;
- redes sociais on-line como o Facebook e Flickr;
- processo de disseminação de vírus e *worms*;
- grafos dinâmicos que representem mobilidade e redes de sensores sem fio.

Esta lista não é exaustiva e dá uma boa idéia do potencial de aplicabilidade da teoria das redes complexas na área de comunicação de dados. O estudo de propriedades não triviais extraídas dos grafos que representam essas redes pode trazer profundas reflexões na área de comunicação de dados e vários esforços têm ocorrido nas medições, análises, modelagem e outros aspectos dessas redes. Este texto trata um pouco desse estudo.

1.1.1. Redes complexas e redes reais

Diversos fenômenos encontrados na natureza podem ser estudados a partir da teoria de redes complexas. Esses sistemas apresentam características em comum como, por exemplo, podem ser modelados por grafos, isto é, vértices e arestas que representam a interconexão entre os diversos elementos desses sistemas. As redes sociais podem ser modeladas dessa maneira de modo que as pessoas são representadas por vértices e as relações que elas estabelecem são representadas por arestas. São exemplos desses sistemas as redes de relacionamento na Internet como o Facebook e Flickr, entre outros, as redes de relacionamentos das organizações e as redes de citações de trabalhos científicos. Da mesma maneira, redes de comunicação como e-mail, redes telefônicas, redes sem fio, Internet e a Web também têm sido estudadas com o mesmo ferramental oferecido pelas redes complexas. Em aplicações em diversas áreas do conhecimento como, por exemplo, as redes de interações entre proteínas, estruturas do cérebro e redes de transportes também são objetos de aplicação da teoria das redes complexas. A vasta amplitude de aplicações em diversas áreas do conhecimento sugere que as redes complexas apresentam um grande potencial a ser explorado e é uma área de pesquisa bastante ativa atualmente (Costa et al. 2007, 2008).

Uma importante característica comum aos sistemas estudados pela teoria das redes complexas, relatada no trabalho de Costa et al. (2007), é que tais sistemas não são completamente aleatórios mas possuem uma arquitetura um pouco mais estruturada. Por exemplo, as topologias de redes que representam as interações entre proteínas e a Internet, que são dois domínios de estudo bastante diferentes, apresentam um padrão de comportamento semelhante: ambas possuem uma estrutura conhecida como "livre de escala" (*scale-free networks*). Dessa maneira, redes formadas por estruturas completamente diferentes apresentam características topológicas semelhantes. Baseados nesse fato, Costa et al. (2007) apresentam um desafio muito interessante para estudo das redes complexas: encontrar as leis fundamentais que geram, modelam e caracterizam tais redes para auxiliar o entendimento do comportamento de sistemas complexos.

No sentido de apresentar as bases para o estudo do comportamento das redes de computadores através das redes complexas o presente minicurso está estruturado como segue. Na seção 1.2 é apresentada uma breve revisão da teoria dos grafos, base do estudo das redes complexas, e a definição da notação empregada no restante deste texto. A seção 1.3 descreve as principais medidas da teoria das redes complexas, úteis para a caracterização de diversos comportamentos dos sistemas complexos. A seção 1.4 define os principais modelos oriundos da teoria das redes complexas. Tais modelos como o *small world* e *scale free* são úteis para a caracterização da geração e do comportamento das redes complexas. A seção 1.5 apresenta o ferramental estatístico necessário para o estudo e análise do comportamento de redes reais através da teoria de redes complexas, bem como diversos exemplos da aplicação dessa teoria no campo das redes de computadores. Por fim, a seção 1.6 traz as considerações finais deste trabalho.

1.2. Revisão da teoria de grafos

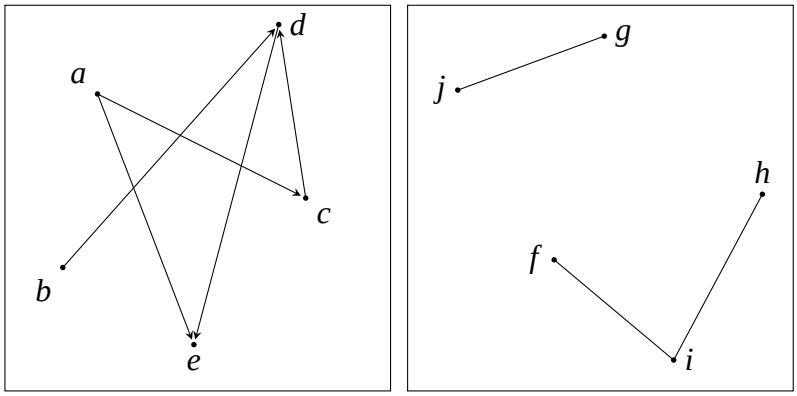
Um grafo é um formalismo matemático que serve para representar objetos e relações entre eles. Esta simples estrutura encontra-se em uma grande diversidade de aplicações como circuitos elétricos, estradas, vários tipos de redes, ecossistemas, relações sociais, interação molecular, bases de dados, estruturas de controle de programas e sistemas de permissões de sistemas de segurança. Estes são alguns exemplos da grande variedade de aplicações da teoria dos grafos. Esta área tem seu foco nas relações de propriedades topológicas de grafos com aquelas derivadas de matrizes de representação ou caracterizações dos mesmos.

Na continuação vamos definir formalmente um grafo, para logo descrever suas características e, finalmente, definir grupos de grafos com propriedades que são de especial interesse para este trabalho.

Um tipo de grafo bastante utilizado na modelagem de redes de computadores é o grafo direcionado simples, definido a seguir.

Definição 1.2.1 (Grafo direcionado simples)**.** Um grafo direcionado simples é uma dupla $\vec{G} = (V, \vec{E})$ onde V é o conjunto finito de vértices, e $\vec{E}$ é o conjunto de arestas. Cada aresta é um par (u, v) de vertices em V, com $u \neq v$.

O termo "simples" indica que não é permitido expressar que um vértice mantenha relacionamento com ele mesmo. Em diversos tipos de redes, como as redes de sensores



(a) Grafo direcionado $\vec{G}_1$ (b) Grafo não direcionado G_2

Figura 1.1. Representação gráfica de grafos simples

sem fio, isto faz muito sentido já que não estamos interessados em modelar autocomunicação pelo canal sem fio. Um caso particular deste tipo de grafo, é o grafo não direcionado simples. A única diferença com o grafo descrito na definição 1.2.1 é que as arestas são representadas como um conjunto, e não como um par ordenado de vértices.

Definição 1.2.2 (Grafo não direcionado simples)**.** Um grafo não direcionado simples é uma dupla $G = (V,E)$ onde V é um conjunto finito de vértices, e E é um conjunto de arestas. Cada aresta é um conjunto $\{u,v\}$ de vertices em V, com $u \neq v$.

No grafo não direcionado simples, cada aresta $\{u,v\}$ não direcionada, representa a presença das arestas direcionadas (u,v) e (v,u) no grafo direcionado simples. A figura 1.1 apresenta estes tipos de grafos. Na figura 1.1(a), o gráfico corresponde ao grafo direcionado $\vec{G}_1 = (V,\vec{E})$ com $V = \{a,b,c,d,e\}$ e $\vec{E} = \{(a,c),(a,e),(b,d),(c,d),(d,e)\}$. Já na figura 1.1(b), o gráfico representa o grafo não direcionado $G_2 = (V,E)$ com $V = \{f,g,h,i,j\}$ e $E = \{\{f,i\},\{g,j\},\{h,i\}\}$.

Uma das representações mais utilizadas para grafos, tanto direcionados quanto não direcionados, é a matriz de adjacência. Este conceito é fundamental para continuar construíndo definições e propriedades sobre grafos.

Definição 1.2.3 (Matriz de adjacência)**.** A matriz de adjacência é uma representação de grafos. Para um grafo $\vec{G} = (V,\vec{E})$, é uma matriz $A(\vec{G})$ de dimensão $|V| \times |V|$, definida por

$$A(\vec{G})[i,j] = \begin{cases} 1 & \text{se} \quad (i,j) \in \vec{E}, \\ 0 & \text{caso contrario.} \end{cases}$$

Nesta definição, e no restante do texto, expressões do tipo "$|V|$" representam a cardinalidade do conjunto V. É importante esclarecer que quando dizemos que a matriz de adjacência é uma representação, estamos afirmando que o grafo $\vec{G}$ e sua matriz de adjacência A contêm a mesma informação, isto é, são equivalentes em conteúdo. Baseados

no grafo da figura 1.1(a), construímos a matriz de adjacência correspondente e obtemos

$$A(\vec{G}_1) = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Neste caso, para indexar, utilizamos como exemplo a função de ordem lexicográfico $I_1(x)\colon V \to \mathbb{N} \cup 0$, isto é $I_1(a) = 0$, $I_1(b) = 1$, $I_1(c) = 2$, $I_1(d) = 3$ e $I_1(e) = 4$. Agora criamos uma nova função de indexação $I_2(x)$, definida por extensão da seguinte maneira: $I_2(f) = 0$, $I_2(g) = 1$, $I_2(h) = 2$, $I_2(i) = 3$ e $I_2(j) = 4$. A matriz de adjacência associada ao grafo da figura 1.1(b) é a seguinte:

$$A(G_2) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}.$$

Para denotar as relações diretas que um vértice mantém com outros, vamos introduzir os conceitos de adjacência e grau de um vértice.

Definição 1.2.4 (Grafo valorado)**.** Grafo valorado é um grafo em que cada aresta tem um valor (peso) associado. Formalmente, um grafo valorado $G = (V,E)$ consiste de um conjunto V de vértices, um conjunto E de arestas e uma função $\Pi\colon E \to \mathbf{P}$.

Utilizamos **P** (conjunto de pesos) de maneira genérica para representar o contradomínio da função Π. O conjunto **P** pode ser convenientemente especificado para o conjunto dos números reais ($\mathbb{R}$) ou inteiros ($\mathbb{Z}$), por exemplo.

Para grafos valorados, a matriz de adjacência pode apresentar valores diferentes de 0 e 1 como apresentado na definição 1.2.3. Para esta situação os valores presentes na matriz serão atribuídos de acordo coma função Π para a respectiva aresta.

Definição 1.2.5 (Subgrafo)**.** Um grafo $H = (V',E')$ é dito ser um subgrafo de $G = (V,E)$ se, e somente se:

- cada vértice de H é também um vértice de G, ou seja, $V' \subseteq V$;
- cada aresta de H é também uma aresta de G, ou seja, $E' \subseteq E$; e
- cada aresta de H tem os mesmos nós terminais em G, ou seja se $(u,v) \in E'$ então $(u,v) \in E$.

Definição 1.2.6 (Adjacência e grau)**.** Em um grafo direcionado $\vec{G} = (V,\vec{E})$, o vertice v é adjacente a u se $A[u,v] = 1$. O número de vértices adjacentes a u chama-se grau de u, e diferenciamos entre grau de saída (*output*)

$$k_u^{\text{out}}(\vec{G}) = \sum_{v \in V\colon v \neq u} A[u,v]$$

e grau de entrada (*input*)

$$k_u^{\text{in}}(\vec{G}) = \sum_{v \in V\,:\, v \neq u} A[v,u].$$

Denotamos o grau de u simplesmente por $k_u(\vec{G}) = k_u^{\text{out}}(\vec{G}) + k_u^{\text{in}}(\vec{G})$.

Definição 1.2.7 (Vizinhança)**.** A vizinhança de um vértice u é definida como $N_u = \{u \mid e_{uv} \in E\}$ e $k_u = |N_u|$.

Os grafos "conexos" apresentam propriedades interessantes para as redes de computadores e para defini-los é necessário, primeiro, introduzir o conceito de caminho simples. Note que ao definir conceitos para grafos direcionados simples, estamos também incluindo a definição para grafos não direcionados, já que os grafos direcionados são mais gerais.

Definição 1.2.8 (Caminho simples)**.** O caminho simples entre v_i e v_j, vértices do grafo direcionado $\vec{G} = (V, \vec{E})$, é qualquer sequência de arestas

$$(v_i, v_{ij_1}), (v_{ij_1}, v_{ij_2}), \ldots, (v_{ij_{n-2}}, v_{ij_{n-1}}), (v_{ij_{n-1}}, v_j),$$

que denotaremos $v_i \rightsquigarrow v_j$. Cada v_{ij} denota um vértice intermediário do caminho desde v_i até v_j. Aqui, o comprimento do caminho é n.

Note que os nós intermediários v_{ij} são irrelevantes para a definição de caminho simples. No sentido mais específico, vamos definir um grafo conexo para grafos não direcionados como segue.

Definição 1.2.9 (Grafo conexo)**.** Um grafo não direcionado simples $G = (V,E)$ é dito conexo se existe um caminho simples $u \rightsquigarrow v$ de qualquer vértice $u \in V$ a qualquer vértice $v \in V$, com $v \neq u$.

Esta definição pode ser aplicada a grafos direcionados simples, mas temos que fazer primeiro uma transformação. Vamos dizer que um grafo direcionado simples $\vec{G}$ é conexo se sua "bi-orientação" é um grafo não direcionado simples G conexo. A bi-orientação é uma função sobre a matriz de adjacência que para cada elemento $A[i,j] = 1$, atribui adjacência para o seu par simétrico, isto é, $A[j,i] = 1$. Outro conceito de interesse é a conectividade forte em grafos direcionados simples, que definimos a seguir:

Definição 1.2.10 (Grafo fortemente conexo)**.** Um grafo direcionado simples $\vec{G} = (V, \vec{E})$ é dito ser fortemente conexo se existe um caminho simples $u \rightsquigarrow v$ de qualquer vértice $u \in V$ a qualquer vértice $v \in V$, com $v \neq u$.

Dentro da teoria espectral de grafos uma definição básica é a de característica polinomial. A partir deste conceito é que vamos definir os autovalores de uma grafo, definição necessária para introduzir o principal conceito desta seção: a conectividade algébrica.

Definição 1.2.11 (Característica polinomial)**.** A característica polinomial de um grafo $\vec{G}$ é o determinante $\det(xI - A(\vec{G}))$, onde I é a matriz identidade de dimensão $|A| \times |A|$, e x uma variável real.

Quando expandimos o determinante da definição 1.2.11, vemos que a característica polinomial é um sistema de equações lineares. Resolvendo esse sistema, podemos achar os autovalores do grafo, definidos a continuação.

Definição 1.2.12 (Autovalores)**.** Os autovalores de um grafo $\vec{G} = (V, \vec{E})$ são as raízes da característica polinomial.

Notemos que para um grafo $\vec{G} = (V, \vec{E})$, existem exatamente $|V|$ autovalores, como veremos abaixo. Os autovalores determinam o espectro do grafo definido da seguinte maneira:

Definição 1.2.13 (Espectro)**.** O espectro de um grafo $\vec{G} = (V, \vec{E})$ é o multiconjunto de autovalores associados. Seja $|V| = m$, então existem m autovalores associados a $\vec{G}$.

O espectro de um grafo fornece caracterizações do mesmo. Mas é por meio do espectro do laplaciano do grafo que obtemos a caracterização de maior interesse neste trabalho: a conectividade algébrica. Com o laplaciano de um grafo é possível determinar se um grafo é conexo apenas com operações matriciais. Porém não se conhece ainda muito sobre as propriedades espectrais do laplaciano de um grafo. Trabalhos recentes nessa área sugerem que há muito mais por descobrir (ver Gross & Yellen 2003, seção 6.5.6). A definição do laplaciano de um grafo é introduzida da seguinte maneira:

Definição 1.2.14 (Laplaciano de um grafo)**.** O laplaciano L de um grafo $\vec{G}$ define-se como $L = D - A(\vec{G})$, onde D é a matriz diagonal de graus dos vertices de $\vec{G}$.

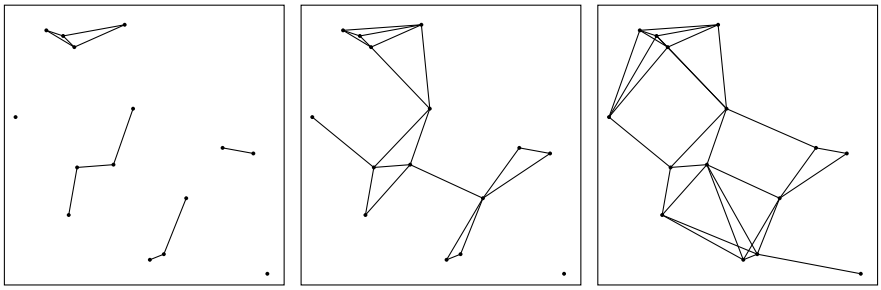
O segundo menor autovalor do laplaciano de um grafo fornece informação a respeito da conectividade do grafo. Antes de conhecer estas propriedades, vamos definir formalmente a conectividade algébrica introduzida por Fiedler (1973).

Definição 1.2.15 (Conectividade algébrica)**.** A conectividade algébrica do grafo $\vec{G} = (V, \vec{E})$ com laplaciano L de autovalores $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ é λ_2.

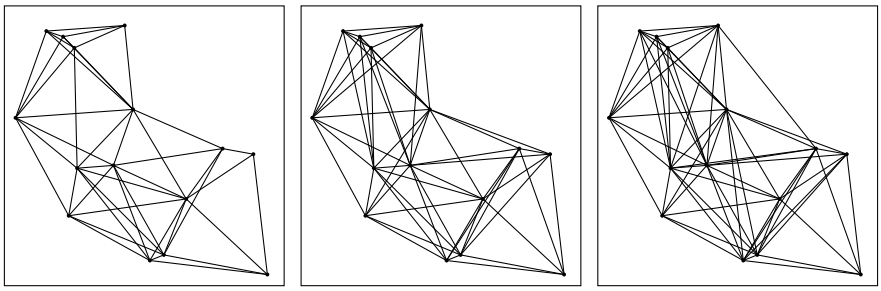
Como discutido acima, a conectividade algébrica é o segundo menor autovalor do laplaciano associado ao grafo. O principal resultado a respeito da conectividade algébrica é apresentado no seguinte teorema.

Teorema 1.2.1. *A conectividade algébrica é positiva ($\lambda_2(L_{\vec{G}}) > 0$) se e somente se o grafo $\vec{G}$ é fortemente conexo.*

Uma característica interessante desse teorema (a prova encontra-se em Fiedler 1973) é que permite determinar se um grafo é conexo ou não apenas com operações matriciais. Para ferramentas como R e Ox, isto representa uma vantagem, já que as operações matriciais estão implementadas com eficiência. Por outro lado, deveríamos analisar o impacto da precisão numérica nos resultados obtidos. Por exemplo, para um grafo conexo, a conectividade algébrica poderia ser 10^{-20}. Um valor negativo muito proximo de zero, implicaria que não teremos classificado o grafo corretamente segundo a propriedade de ser conexo ou não.



(a) $\lambda_2 = -1,591 \cdot 10^{-16}$ (b) $\lambda_2 = 0$ (c) $\lambda_2 = 0,486$



(d) $\lambda_2 = 1,199$ (e) $\lambda_2 = 2,223$ (f) $\lambda_2 = 2,898$

Figura 1.2. Comportamento da conectividade algébrica

Porém a conectividade algébrica tem um significado mais forte. Quanto menor o seu valor, "mais longe" está o grafo de ter conectividade, e quando o valor cresce passando a ser positivo, quanto maior for, "mais forte" é sua conectividade, isto é, a eliminação de algumas arestas do grafo não tira a propriedade do mesmo ser conexo. A figura 1.2 mostra o comportamento da conectividade algébrica segundo o grafo.

1.3. Caracterização de redes complexas

Nesta seção serão apresentadas diversas medidas que tipicamente são utilizadas nos estudos que envolvem a teoria das redes complexas. Essas medidas são capazes de expressar as características topológicas mais relevantes das redes. Neste trabalho foi feita uma classificação das medidas em: medidas relacionadas à distância, medidas de centralidade, medidas espectrais, medidas egocêntricas, medidas de identificação de comunidades e medidas de hierarquia. A lista aqui apresentada não esgota todas as medidas presentes na literatura mas dá uma boa visão das medidas mais frequentemente utilizadas. O trabalho de Costa et al. (2007) é uma referência bastante abrangente sobre esse tema.

1.3.1. Medidas relacionadas à distância

De uma maneira geral, distância está associada a quantidade de arestas contidas em um caminho que conecta dois vértices u e v, denotada por $d_{u,v}$. Essa definição é válida para grafos não valorados. Observe que para grafos dirigidos em geral $d_{u,v} \neq d_{v,u}$, dado que o caminho em um sentido é, em geral, diferente do caminho no sentido oposto. Um caminho geodésico, ou caminho mínimo (*shortest path*) é qualquer caminho que interligue os vértices u e v e tenha distância mínima, denotada por d_G. Nota-se que podem existir

diversos caminhos mínimos entre quaisquer dois vértices em um grafo. O conceito de distância pode ser facilmente estendido para grafos valorados, levando em consideração o peso associado às arestas. Quando não há um caminho entre dois vértices u e v atribui-se distância infinita, ou seja, $d_{u,v} = \infty$.

Uma medida relacionada à distância que aparece com muita frequência no contexto de redes complexas é o comprimento médio dos caminhos mínimos, ou, em inglês, *average path length* (L) do grafo $G = (V, E)$, definido por:

$$L = \frac{1}{n(n-1)} \sum_{i \in V} \sum_{j \in V} SP_{ij},$$

onde $n = |V|$ é o número de vértices do grafo e SP é o comprimento do caminho geodésico entre os vértices i e j.

Para evitar que esta soma divirja, ou seja, a soma acima ir para infinito, apenas vértices que estão conectados são contabilizados. Mais formalmente, $i, j \in V'$, onde $V' = \{u \in V \mid \exists u \rightsquigarrow v \in V, u \neq v\}$. Esta medida tem uma importância destacada na teoria de redes complexas pois é utilizada para caracterização das redes *small world*.

Algumas outras medidas estão relacionadas a caminhos geodésicos mas, neste trabalho, foram consideradas como medidas de centralidade que estão apresentadas na próxima seção.

1.3.2. Medidas relacionadas a agrupamentos e ciclos

Nesta seção serão apresentadas algumas medidas que caracterizam a tendência de algumas redes reais a formarem conjuntos de vértices agrupados.

Uma medida que caracteriza agrupamento e que possui um destaque especial na teoria de redes complexas é o coeficiente de agrupamento (*clustering coefficient*) para um vértice u, também conhecido por transitividade. Para grafos não dirigidos, é definido como

$$CC_u = \frac{2|\{e_{vw}\}|}{k_u(k_u - 1)},$$

onde $v, w \in N_i$, $e_{vw} \in E$ e $k_u(k_u - 1)/2$ é o número máximo de arestas entre vértices na vizinhança de u. Essa medida também pode ser utilizada na caracterização da propriedade *small world*.

O coeficiente de agrupamento da rede é então definido como

$$CC = \frac{1}{|V|} \sum_{u \in V} CC_u.$$

Para grafos dirigidos, o coeficiente de agrupamento é um conceito mais complexo e foge do escopo desse trabalho. Algumas propostas para tal estão presentes no trabalho de Costa et al. (2007).

Uma outra medida é o coeficiente do "clube dos ricos" (*Rich-club coefficient*). Essa medida foi proposta por Zhou & RJ (2004) ao perceberem que a topologia da Internet obedece a um padrão que os *hubs* estão bem conectados entre si. Esse mesmo padrão em

redes de citação científica onde pesquisadores influentes tendem a formarem grupos e publicarem artigos conjuntamente. A medida proposta por Zhou & RJ (2004) captura esse comportamento e é definida como

$$\phi(g) = \frac{1}{|\mathscr{R}(g)|(|\mathscr{R}(g)|-1)} \sum_{i,j \in \mathscr{R}(g)} a_{i,j},$$

onde $\mathscr{R}(g)$ é o conjunto que denota o "clube dos ricos" de grau g na rede representada pelo grafo $G = (V,E)$, ou seja, $\mathscr{R}(g) = \{v \in V \mid k_v > g\}$, e a_{ij} é o valor correspondente aos vértices i e j na matriz de adjacência de G. Essa medida é semelhante ao coeficiente de agrupamento para a fração de vértices que apresentam grau maior que g.

1.3.3. Medidas de centralidade

Métricas de centralidade são utilizadas para quantificar a intuição de que, sob algum aspecto, as redes possuem elementos posicionados mais ao centro de sua estrutura do que outros. Na teoria dos grafos, a idéia de classificar um vértice por sua centralidade estrutural foi introduzida em 1948 por Alex Bavelas, resultando na proposição do primeiro índice de centralidade para redes conectadas, o índice de Bavelas.

Medidas de centralidade têm o intuito de estimar a importância de um dado vértice, ou seja, ranqueá-lo segundo sua importância topológica. Nós em posições centrais geralmente têm grande importância estrutural e, em casos onde há fluxo de dados, são agentes de escoamento vitais. Geralmente, quanto maior for a participação de um vértice ou de uma aresta em caminhos no grafo, maior será sua importância. Desta forma, pode-se saber o quão importante é, por exemplo, um elemento computacional para uma rede, ou uma pessoa em uma rede social, inferindo-se esta importância a partir da posição desta entidade frente às demais. É importante ressaltar que o termo centralidade está intimamente ligado à perspectiva das características consideradas pela métrica, pois, por exemplo, um elemento pode ser considerado mais central por estar ligado a outros elementos importantes, ou por estar mais próximo de todos os outros nós, ou ainda por estar presente em muitos caminhos quando se percorre o grafo.

Existem diversas métricas de centralidade baseadas em diferentes características do grafo como, por exemplo, os conceitos de distância entre nós, ou do grau do nós são amplamentes explorados pelas métricas *closeness*, *degree centrality*, *eccentricity* e *centroid*. Além disto, outra característica base usada pelas métricas de centralidade são os menores caminhos do grafo, utilizados, por exemplo, pelo *shortest-path betweeness centrality*, também conhecido como *betweenness*, *stress centrality* e *reach*. Existem ainda métricas que utilizam a importância dos vértices de sua vizinhança para o cálculo da importância própria como é o caso da métrica *eigenvector*, ou ainda as métricas correlatas de classificação de páginas da *World-Wide Web hub Score* e *authority*. Ainda no contexto da *Web* uma métrica de centralidade amplamente estudada é o *PageRank*, que é a base da máquina de busca do Google.

A utilização de métricas de centralidade no projeto de algoritmos no contexto de comunicação em redes de computadores vem sendo assunto de estudos recentes na literatura como, por exemplo, a possibilidade de se utilizar a centralidade como forma de se balancear carga e assim evitar a sobrecarga de nós mais centrais. A centralidade

também pode ser utilizada para diminuir a latência empregando um tráfego de dados ciente da centralidade dos nós (Krause et al. 2006).

1.3.3.1. Distância e vizinhança

Nesta seção serão apresentados índices que ranqueiam os nós por sua centralidade baseado em características básicas como a noção de vizinhança e distância dentro de um grafo.

Degree centrality É o índice de centralidade mais simples e, para grafos não direcionados, é representado como $C_D(v)$, que corresponde ao número de arestas ligadas ao vértice v. O *degree centrality* é usualmente apresentado de maneira normalizada, pois, desta forma, pode-se ter uma noção do quão distante este nó está das centralidades mínimas e máximas. Para tanto, a versão normalizada é calculada por $C_{Dn}(v) = k(v)/n-1$, em que $k(v)$ é o grau do vértice v e n é o número de vértices do grafo. Para grafos direcionados, a métrica se divide de maneira a representar o grau de entrada, *in-Degree centrality* $C_{Di}(v)$ e o grau de saída, *out-Degree* centrality $C_{Do}(v)$. O *degree centrality* é uma métrica local, ou seja, é necessário apenas o conhecimento da quantidade de nós da vizinhança de primeiro nível (nós diretamente conectados) para que esta métrica seja calculada.

Eccentricity Esta métrica é definida como a distância máxima entre um vértice $u \in V$ e qualquer outro vértice $v \in V$ da rede, i.e,

$$Ecc(u) = \max_{v \in V} d_G(u,v).$$

Esta métrica pode ser empregada em um típico problema de localização, encontrar um vértice cuja distância máxima para todos os outros nós é mínima. Tal problema pode ser solucionado determinando-se o vértice u tal que $Ecc(u)$ seja mínimo. Em outras palavras, o diâmetro do grafo é o valor máximo de Ecc enquanto o raio é o valor mínimo.

Closeness Inicialmente, é importante denotar o somatório das distâncias geodésicas de um vértice $u \in V$ para um outro vértice qualquer $v \in V$, i.e., $\sum_{v \in V \setminus u} d_G(u,v)$. Este conceito de somatório das distâncias é empregado, por exemplo, em problemas de localização. O problema de localização mini-soma consiste na idéia de se minimizar a distância total percorrida num sistema de distribuição, ou seja, uma abordagem para a solução deste problema de localização é encontrar o conjunto de vértices com o menor valor de *closeness*.

O *Closeness* de um vértice u cresce à medida que o total da soma das distâncias diminui e é definido como a recíproca da soma das distância geodésicas para todos os outros vértices $v \in V$, ou seja,

$$C_{CL}(u) = \frac{1}{\sum_{v \in V \setminus u} d_G(u,v)}.$$

Esta métrica apresenta um problema quando aplicada em grafos desconexos, pois a distância entre vértices de componentes distintos é, normalmente, definida como infinito. Desta forma, o valor do *closeness* $1/\infty$ para todos os vértices será o mesmo

Em Valente & Foreman (1998) são propostas duas métricas baseadas no *closeness*: *integration*, que mede o quão bem conectado um membro da rede está e *radiality*, que mensura o quanto as ligações de um nó expandem-se pela rede. Vértices com *integration* alto ficam, mais cedo, cientes de informações trafegadas pela rede pois, na média, eles estão mais próximos dos demais vértices, enquanto vértices com valor alto de *radiality* supostamente são bons emissores de informação.

1.3.3.2. Caminhos mínimos

Nesta seção serão mostrados índices de centralidade que fazem uso dos caminhos mínimos de um grafo para definir o valor de centralidade de um vértice. Caminhos mínimos são usualmente definidos para vértices, mas podem ser aplicados também para arestas, tanto que algumas métricas que existiam apenas para centralidade de vértice, mas posteriormente foram propostas para considerar centralidade de aresta.

Stress centrality Esta é a métrica mais simples que faz uso da enumeração dos caminhos mínimos de um grafo, originalmente proposta em Shimbel (1953) para mensurar a quantidade de trabalho que um elemento precisa suportar numa rede. Para um vértice u, esta métrica é denotada pelo número de caminhos mínimos que contém u, exceto aqueles que comecem, ou terminem em u. Formalmente, o *stress centrality* para u é definido como

$$C_S(u) = \sum_{s \neq u \in V} \sum_{t \neq u \in V} \sigma_{st}(u),$$

onde $\sigma_{st}(u)$ é o número de caminhos mínimos que contém o vértice u. Analogamente, esta métrica pode ser definida para uma aresta e qualquer como:

$$C_S(e) = \sum_{s \in V} \sum_{t \in V} \sigma_{st}(e),$$

onde $\sigma_{st}(e)$ é o número de caminhos mínimos que contém a aresta e.

Shortest-path Betweeness Centrality Esta métrica, usualmente chamada apenas de *betweeness* foi proposta em Anthonisse (1971) e Freeman (1977). Basicamente é uma variação do *stress centrality* com o acréscimo da visão geral de todos os caminhos mínimos presentes no grafo. Formalmente, para um vértice u, esta métrica é definida como:

$$C_B(u) = \sum_{s \neq u \in V} \sum_{t \neq u \in V} \frac{\sigma_{st}(u)}{\sigma_{st}},$$

onde σ_{st} representa a quantidade de caminhos mínimos entre dois vértices quaisquer s e t, enquanto $\sigma_{st}(u)$ representa a quantidade de caminhos mínimos em σ_{st} que passem pelo

vértice u. Assim como no *stress centrality*, os caminhos que comecem ou terminem em u são excluídos da contagem.

Ao contrário do *closeness*, o *betweenness* não possui problemas ao ser aplicado em grafos desconexos, pois a fração referente a qualquer par de vértices s e t para os quais não existe caminho será considerada 0.

O algoritmo mais eficiente conhecido o cálculo do *betweenness* de um grafo possui complexidade de espaço $O(n+m)$ e de tempo $O(nm)$ para grafos não valorados e $O(nm+n^2 \log n)$ para grafos valorados, apresentado no trabalho de Brandes (2001).

Portanto, o cálculo exato desta métrica pode se tornar uma tarefa impraticável para grafos com milhões de nós e, desta forma, para estes casos pode-se trocar precisão no resultado da métrica por uma diminuição do custo computacional para o cálculo da mesma. Para certas aplicações, o valor exato da centralidade de cada vértice não é importante, contanto que os vértices mantenham-se ranqueados por centralidade da mesma forma como estariam se o cálculo exato fosse feito. Em Brandes & Pich (2007) é proposta a idéia do uso de nós pivô como forma de se atenuar o custo computacional para o cálculo do *betweenness*. Ao contrário do algoritmo proposto em Brandes (2001) em que cada um dos nós precisa fazer uma busca em profundidade, apenas um pequeno conjunto de pivôs executam a busca e o resultado é extrapolado com a ajuda de métodos estatísticos, que posteriormente são melhorados em Geisberger et al. (2008).

1.3.3.3. Medidas aplicadas à Web

Devido ao imenso tamanho da World-Wide Web (WWW), um problema considerável é entregar a um usuário aquilo que mais se adequa aos seus interesses quando o mesmo faz uma busca. A modelagem através de grafos se ajusta à estrutura da WWW, uma vez que nós são as páginas e as arestas são modeladas como os *hiperlinks*. Desta forma, a máquina de busca pode escolher o resultado que é mais apropriado à pesquisa de um usuário a partir de um índice de centralidade que ranqueia os vértices do grafo da WWW. Nesta seção serão apresentados os índices *PageRank* e *hubs & authorities*.

PageRank Intuitivamente, o *PageRank* se baseia num modelo do comportamento de usuário que navega pelas páginas e, de vez em quando, desiste daquela e vai buscar outra página diferente. A probabilidade de desistência é o chamado *damping factor*. Este índice, proposto em Brin (1998), leva em consideração apenas a característica topológica da rede, ou seja, uma página terá sua importância definida apenas pela posição que ela ocupa no grafo. O *PageRank* é uma métrica que é definida baseada na centralidade dos vizinhos e em sua quantidade.

Inicialmente todas as páginas começam com um valor de centralidade igualmente definido entre 0 e 1. Desta forma, sem considerarmos o modelo de navegação do usuário, o *PageRank* de uma página u é:

$$PR(u) = \frac{1-d}{N} + d \sum_{v \in B_u} \frac{PR(v)}{L(v)},$$

em que d é o *damping factor*, N é o número de páginas, B_u é o conjunto de páginas que tem *links* apontando para u e $L(v)$ é o número de *hiperlinks* que apontam para v. Vários estudos já foram feitos para avaliar o comportamento do algoritmo variando-se o *damping factor*, mas geralmente assume-se seu valor como 0.85.

Hubs & Authorities Proposto em Kleinberg (1999) e usualmente chamado de *HITS*, este índice atribui dois valores a um vértice referentes às grandezas propostas, *hub* e *authority*. Diferentemente do *PageRank*, ele considera o conteúdo da página aliado à forma com que a mesma está ligado às outras no cálculo destas grandezas. Segunda a definição do autor, um bom *hub* é aquele que aponta para para muitos *authorities* bons, enquanto um bom *authority* é aquele apontado por muitos *hubs* bons

Inicialmente, todos as páginas começam com os valores de *hub* e de *authority* igual a 1. Uma vez que o cálculo do *authority* depende do valor *hub* e vice-versa, o algoritmo atualiza cada um destes valores para uma página u da seguinte forma:

$$C_{Auth}(u) = \sum_{i=1}^{n} C_{Hub}(i),$$

em que i é uma página conectada a u e n é o número total de páginas conectadas a u. Desta forma, o valor *authority* de uma página é a soma de todos os valores *hub* das páginas que apontam para ela. Da mesma maneira,

$$C_{Hub}(u) = \sum_{i=1}^{n} C_{Auth}(i),$$

em que i é uma das páginas a qual u se conecta e n é o número total de páginas a que u se conecta. Desta forma, o valor *hub* é a soma dos valores *authority* das páginas a que ela se liga. O algoritmo obtém valores que convergem após infinitas execuções (Kleinberg 1999) de cada um dos somatórios mostrados acima, um após o outro, sendo necessário, ao final, normalizar a matriz de adjacências do grafo original.

1.3.4. Medidas egocêntricas

Redes egocêntricas, ou redes de vizinhança são redes constituídas apenas de um elemento conhecido como *ego* e aqueles aos quais ele está ligado, i.e, seus vizinhos chamados *alters*. Informações tais como a maneira com que os *alters* estão conectados podem ser coletadas pelo *ego*.

A teoria que dá suporte ao conceito de redes egocêntricas supõe que a estrutura composta por *ego* e *alters* pode ser amostrada de um grande conjunto de elementos e, os resultados de análises sobre a amostra podem ser generalizados para toda a rede, tornando esse estudo atraente. Porém, se por um lado a utilização de uma pequena parte da rede composta apenas de um vértice (*ego*) e de sua vizinhança (*alters*) embute simplicidade, por outro lado, a análise dos resultados fica usualmente restrita à conectividade, ou densidade (Burt 1995).

Um ramo de estudos das redes egocêntricas é a proposição e aplicação de métricas que considerem esses aspectos. Métricas como, por exemplo, *eigenvector* e o *closeness*

não são aplicáveis uma vez que elas avaliam a interação entre um *alter* e os demais elementos de toda a rede, desconsiderando assim o conjunto restrito à rede egocêntrica.

Um exemplo de uma métrica que utiliza o conceito das redes egocêntricas é apresentado a seguir.

Ego-Betweenness Métrica proposta em Everett & Borgatti (2005) que, para determinados tipos de grafos, apresenta alta correlação com o *shortest-path betweenness*, mas que apresenta custo computacional significativamente menor. Embora não quantificada no artigo, o autor mostra a existência de correlação entre o valor do *betweenness* de um vértice e o valor em sua versão egocêntrica, para este mesmo vértice.

Ao contrário do *shortest-path betweenness*, a sua versão egocêntrica é computacionalmente muito mais simples de se calcular. Para cada nó i, extrai-se a matriz de adjacências A que contém i juntamente com seus vizinhos imediatos, i.e., aqueles vértices que estão a um salto de i. Portanto, os caminhos mínimos contidos no grafo induzido por esta matriz terão tamanhos um, que ligam o *ego* aos *alters* e dois, que ligam dois *alters* passando pelo *ego*. Entre cada par de nós não adjacentes nesta matriz existe um caminho mínimo de tamanho 2, que serão os únicos considerados no cálculo desta métrica. Para se extrair o número de caminhos de tamanho dois entre i e j faz-se $A^2\,[1-A]_{i,j}$. O somatório das posições da matriz que não possuírem zero, divido por dois, é o valor do *ego-betweenness* de i.

1.4. Modelos de redes complexas

A observação de padrões comuns no comportamento de redes, mesmo as de natureza diferentes, estimulou diversos pesquisadores a desenvolverem modelos que descrevem e caracterizam tais comportamentos. Nesta seção são apresentados os modelos que frequentemente aparecem nos estudos das redes de computadores como os grafos aleatórios, *small world*, *scale free*, comunidades e as redes geográficas.

1.4.1. Grafos aleatórios

Um dos modelos de redes mais simples e antigo é o modelo de grafos aleatórios Bolobás (2001), Janson et al. (1999). Grafos aleatórios foram inicialmente estudados por Solomonoff e Rapoport (Solomonoff & Rapoport 1951) e extensivelmente estudado por Paul Erdős e Alfréd Rényi (Erdős & Rényi 1959, 1960). Erdős e Rényi estudaram dois diferentes modelos de grafos aleatórios. Esses dois modelos são normalmente diferenciados pela sua notação: $G_{n,m}$ e $G_{n.p}$:

- $G_{n,m}$ é o conjunto de todos os grafos aleatórios que consistem de n nós e m arestas. Para gerar uma amostra de um grafo aleatório pertencente ao conjunto $G_{n,m}$ basta adicionar m arestas aleatórias, uniformemente distribuídas, entre os n vértices inicialmente desconexos.

- $G_{n,p}$ é o conjunto de todos os grafos aleatórios que consistem de n vértices, onde cada par de vértices é conectado de acordo com uma probabilidade p. Para gerar

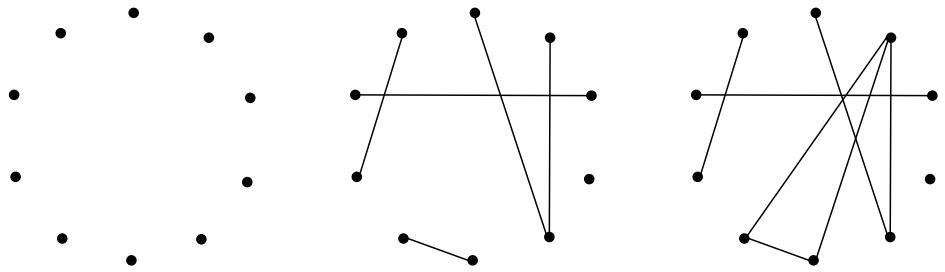
um grafo aleatório segundo esse modelo, adiciona-se arestas entre todos os pares de vértices inicialmente desconexos segundo uma probabilidade p.

É importante observar que na geração de grafos aleatórios segundo o modelo $G_{n,m}$, a quantidade de arestas no grafo é fixa. Entretanto, a quantidade de arestas na criação de um grafo aleatório segundo o modelo $G_{n,p}$ pode variar, mas em média, o valor é o mesmo para n vértices e a probabilidade p. É importante observar que em ambos os modelos, dependendo da quantidade de arestas a ser adicionada no grafo ou da probabilidade de criação de arestas entre dois vértices, o grafo resultante pode ser desconexo. Além disso, grafos aleatórios $G_{n,p}$ possuem uma distribuição de graus binomial. Ou seja, a probabilidade p_k que um vértice qualquer seja conectado por exatamente k outros vértices é

$$p_k = \left(\begin{array}{c} n \\ k \end{array} \right) p^k (1-p)^{n-k}.$$

Entretanto, dependendo da aplicação, é necessário saber quantas arestas são criadas no grafo inicialmente desconexo. Para isso, seja n a quantidade de vértices e p a probabilidade de criação de arestas, a quantidade de arestas criadas no grafo é, em média, $pn(n-1)/2$. Uma característica importante de grafos aleatórios é o pequeno comprimento médio dos caminhos mínimos (L) entre os pares de nós. Isso acontece porque as arestas são criadas no grafo de maneira aleatória. Entretanto, o coeficiente de agrupamento do grafo é, tipicamente, pequeno.

A figura 1.3 ilustra a criação de um grafo aleatório para diferentes valores de probabilidade. Na figura 1.3(a), quando a probabilidade de criação de arestas é 0, nenhuma aresta é criada no grafo. Quando a probabilidade de criação de arestas é 0.1 ou 0.15 (Figura 1.3(b) e Figura 1.3(c), respectivamente) as arestas são criadas de maneira aleatória entre os vértices. É interessante observar que se $p < 1$, o grafo resultante pode ser desconexo.



(a) $p = 0$ (b) $p = 0.1$ (c) $p = 0.15$

Figura 1.3. Exemplo de criação de um grafo aleatório segundo $G_{n,p}$

1.4.2. *Small world*

Redes *small world* recebem esse nome por analogia ao fenômeno *small world*, também conhecido como “seis graus de separação”. O fenômeno *small world* corresponde à hipótese de que em uma rede social a distância para conectar duas pessoas quaisquer em qualquer lugar do mundo é, em geral, pequena. Em 1967, Stanley Milgram testou

experimentalmente essa hipótese e descobriu que, em média, apenas 6 pessoas separam quaisquer pessoas no mundo (Milgran 1967).

Estudos recentes definem uma forma de mensurar redes com características *small world* de forma genérica (Watts & Strogatz 1998). Para uma rede ser chamada de *small world*, ela deve possuir as seguintes características: (i) pequeno comprimento médio de caminhos mínimos entre nós e (ii) alto coeficiente de agrupamento. Entretanto, como mensurar o que é um pequeno comprimento médio de caminhos mínimos ou alto coeficiente de agrupamento? Para isso, devemos criar um grafo aleatório com as mesmas características do grafo original, em termos de número de nós e arestas. Após isso, encontre as métricas de caminho médio e mínimo e coeficiente de agrupamento de ambos os grafos. Uma rede é considerada *small world* se o comprimento médio de caminhos mínimos é próximo em relação ao grafo aleatório equivalente, entretanto, possuindo valores de coeficiente de agrupamento muito maior.

De maneira formal, seja CC e L os valores de coeficiente de agrupamento e caminho médio mínimo da rede que deseja-se saber se possui características *small world*. Além disso, seja CC_a e L_a as mesmas variáveis para o grafo aleatório equivalente ao grafo anterior. Para uma rede possuir características *small world*, as seguintes inequações devem ser verdadeiras: $CC \gg CC_a$ e $L/L_a \sim 1$. A Tabela 1.1 ilustra o caminho médio mínimo e o coeficiente de agrupamento de diferentes redes *small world* em comparação com grafos aleatórios gerados com as mesmas características das redes.

Na literatura de redes complexas, existem vários modelos de geração de redes com características *small world*. Como exemplo temos os modelos de Watts & Strogatz (Watts & Strogatz 1998), Newman & Watts (Newman & Watts 1999) e o modelo de kleinberg (Kleinberg 2000). Para gerar uma rede com características *small world* nesses modelos, parte-se de um grafo regular (grafos em que os nós apresentam grau constante). Grafos regulares apresentam altos comprimentos médios dos caminhos mínimos e coeficiente de agrupamento. A idéia é adicionar atalhos na rede, diminuindo o comprimento médio dos caminhos mínimos entre os nós mantendo altos valores de coeficiente de agrupamento.

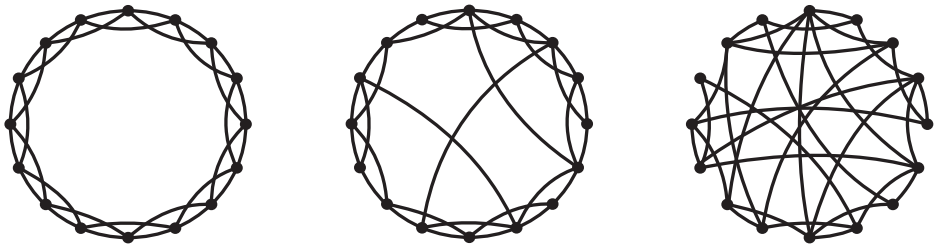
Modelo de Watts & Strogatz O primeiro modelo para criação de uma rede com características *small world* foi proposto por Watts & Strogatz. Nesse modelo, parte-se de um grafo regular e, para cada aresta do grafo, a aresta em questão é reposicionada no grafo de maneira aleatória segundo uma probabilidade pré-definida, mantendo um dos seus pontos finais. Dessa forma, a quantidade de arestas do grafo resultante é a mesma. Entretanto, o grau de cada vértice se altera. Os autores mostraram que para valores pequenos de prob-

Tabela 1.1. Exemplos empíricos de redes small world Watts & Strogatz (1998)

	L	L_a	CC	CC_a
Atores	3.65	2.99	0.79	0.00027
Rede de Energia	18.7	12.4	0.080	0.005
C. elegans	2.65	2.25	0.28	0.05

abilidade uma rede regular pode ser transformada em uma rede com características *small world*. Quando a probabilidade de reposicionamento das arestas é grande, o grafo regular é transformado em um grafo aleatório, possuindo pequenos valores de comprimento médio dos caminhos mínimos e coeficiente de agrupamento.

A Figura 1.4 ilustra a criação de uma rede com características *small world*. Quando $p = 0$ (Figura 1.4-(a)), nenhuma a resta do grafo original é reposicionada e têm-se um grafo regular. Quando a probabilidade de reposicionamento de arestas possui valores $0 < p \ll 1$ (Figura 1.4-(b)), pode-se perceber que algumas arestas do grafo foram reposicionadas, servido como atalhos na rede. Nesse caso, o caminho médio mínimo entre os nós diminui, mantendo altos valores de coeficiente de agrupamento. É importante observar que quando uma aresta é reposicionada, um de seus pontos finais é mantido e o outro, é reposicionado de maneira aleatória na rede. Quando a probabilidade de reposicionamento de arestas é próxima de 1, o grafo apresenta características de grafos aleatórios, devido ao grande número de arestas reposicionadas (Figura 1.4-(c)).



(a) $p = 0$ (b) $0 < p \ll 1$ (c) $0 \ll p \leq 1$

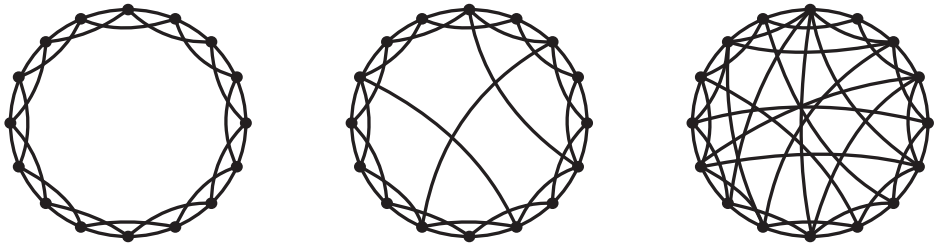
Figura 1.4. Modelo de Watts & Strogatz na criação de uma rede com características small world.

Modelo de Newman & Watts No modelo anterior, para gerar um rede com características *small world*, as arestas do grafo eram reposicionadas segundo uma probabilidade p. Entretanto, em alguns tipos de redes, o reposicionamento de arestas não se aplica. Dessa forma, Newman & Watts propuseram um modelo que, ao invés de reposicionar as arestas do grafo segundo uma probabilidade, novas arestas são adicionadas ao grafo. Nesse modelo, para cada aresta existente no grafo regular, uma nova aresta é adicionada segundo uma probabilidade p. Além disso, um dos pontos finais da aresta adicionada é o mesmo de um dos pontos finais da aresta que foi verificada. Os autores mostraram que para valores intermediários da probabilidade de adição de arestas, a nova rede criada também apresenta características *small world*.

Nesse modelo, a quantidade de arestas no grafo aumenta de acordo com a probabilidade utilizada. Por exemplo, seja $p = 0.1$ a probabilidade de adição de arestas. Se o grafo original possuir 100 arestas, o grafo resultante possuirá, em média, 110 arestas. As 10 arestas adicionadas serão os atalhos na rede, diminuindo o caminho médio mínimo entre os nós mantendo altos valores de coeficiente de agrupamento. É importante observar que quando a probabilidade de adição de arestas é 1, a rede resultante possuirá o dobro de aresta, já que para cada aresta do grafo uma nova aresta será criada.

A Figura 1.5 ilustra a adição de arestas no modelo de Newman & Watts. Quando

$p = 0$, nenhuma aresta é adicionada no grafo original (Figura 1.5(a)). Quando o valor de p é maior do que 0, o modelo adiciona algumas arestas no grafo original, transformando-o em um grafo com características small world. Quando o valor de p é próximo de 1, o grafo original é transformado em um grafo aleatório. É importante observar que para todos os valores de probabilidade, as arestas segundo o modelo de Newman & Watts são adicionadas no grafo original, e não reposicionadas.



(a) $p = 0$ (b) $0 < p \ll 1$ (c) $0 \ll p \leq 1$

Figura 1.5. Modelo de Newman & Watts na criação de uma rede com características small world.

Modelo de Kleinberg Os modelos vistos anteriores para reposicionamento ou adição de arestas seguem uma probabilidade fixa p para todas as arestas do grafo original, independente se o novo ponto final da aresta que será adicionada no grafo está perto ou longe da aresta verificada. Kleinberg propôs um modelo onde a distância entre os nós é um fator para a criação de atalhos no grafo. Essa distância pode ser entendida como distância euclidiana ou distância em saltos entre os dois nós em questão. Quando a distância em número de saltos for utilizada, será utilizado a quantidade de saltos no caminho mínimo entre os dois nós. Dessa forma, segundo o modelo de Kleinberg, uma aresta é adicionada na rede segundo uma probabilidade $p = 1/d_{i,j}^r$, onde $d_{i,j}$ é a distância entre os dois nós e r é um parâmetro estrutural na criação de novas arestas, ou seja, o valor de r definirá a importância da distância na criação de uma arestas entre os nós i e j. Dessa forma, para cada aresta existente no grafo original, um novo vértice escolhido de maneira aleatória será avaliado por $p = 1/d_{i,j}^r$ e, em caso afirmativo, uma nova aresta será criada na rede.

É importante observar que o modelo de Kleinberg necessita de dois parâmetros na definição na adição de uma aresta na rede, $d_{i,j}$ e r. Dessa forma, se dois nós estão distantes mas o valor de r é próximo de zero, a probabilidade de criação de uma aresta entre eles é alta. Entretanto, se o valor de r for suficientemente maior do que 1, a probabilidade será baixa. É interessante observar que o valor de r pode ser entendido como o custo para a criação de um atalho na rede. Se dois nós estão geograficamente ou em número de saltos distantes, a criação de um atalho entre eles terá um custo maior se a ligação entre eles for feita, por exemplo, de maneira cabeada.

1.4.3. *Scale free*

As Redes Livre de Escala (RLEs) (Newman 2003, Li et al. 2005, Keller 2005), ou *Scale-free Networks*, foram introduzidas por Barabási e Albert em 1999 (Barabási & Albert 1999) para modelar topologias de redes em que a distribuição da conectividade dos nós

segue uma lei de potência (Clauset et al. 2007). Desde então, podem ser verificados na literatura diversos outros trabalhos que modelam os mais variados sistemas, sejam naturais ou tecnológicos, como RLEs. Como exemplos de redes livre de escala, pode-se citar a Internet (Faloutsos et al. 1999), a *World Wide Web* (Albert et al. 1999), serviços de redes sociais online (Leskovec et al. 2008), redes de colaboração científica (Newman 2001), cadeias alimentares (Camacho et al. 2002), redes de contatos sexuais (Lilijeros et al. 2001) e rotas ferroviárias (Faloutsos et al. 1999).

A distribuição da variável aleatória X que define a conectividade dos nós de uma rede segue uma lei de potência se, dado a sua função distribuição acumulada $F(x) = P(X \leq x)$ e a sua função distribuição acumulada complementar $\overline{F}(x) = P(X > x)$, $\overline{F}(x) = 1 - F(x) \approx cx^{-\alpha}$ para alguma constante $0 < c < \infty$ e índice de cauda, ou *tail index*, $\alpha > 0$ (Li et al. 2005). Para $1 < \alpha < 2$, F tem variância infinita e média finita mas, para $0 < \alpha \leq 1$, tanto a variância quanto a média são infinitas. Além disso, uma propriedade interessante dessa distribuição é que $\log P(X > x) \approx \log(c) - \alpha \log x$, fazendo com que o gráfico de $\overline{F}$ em escalas logarítmicas seja uma reta com inclinação $-\alpha$ para altos valores de x. Para exemplos e maiores detalhes sobre como identificar leis de potência em dados empíricos, consulte Clauset et al. (2007).

1.4.3.1. Modelos

Por serem redes particulares, com características que não podem ser reproduzidas por modelos de redes aleatórias como, por exemplo, o modelo de Erdős & Rényi (1960), as RLEs demandam que outros modelos sejam propostos. Assim, Barabási & Albert (1999) propuseram o primeiro modelo que explica tais características. Esse modelo, comumente chamado de modelo Barabási-Albert ou simplesmente modelo BA, incorpora duas novas características para gerar a rede: crescimento (*growth*) e conexão preferencial (*preferential attachment*). A primeira estipula que o número de nós da rede cresce com o tempo, enquanto a segunda estipula que quanto mais conectado for um nó, maiores são as chances dele receber novas conexões. Esse fenômeno, também conhecido como "ricos ficam mais ricos", vai fazer com que a rede tenha poucos nós com muitas conexões, também chamados de *hubs*, e muitos nós com poucas.

A geração da rede segundo o modelo BA se dá a partir de uma rede conexa com $n_0 > 2$ nós iniciais $v_1, ..., v_{n_0}$. Logo sem seguida, um novo nó v_{n_0+1} é adicionado à rede e conecta-se a um nó $v_j, 0 < j \leq n_0$ de acordo com uma probabilidade que é proporcional ao grau d_j de v_j. Formalmente, a probabilidade $p_{i,j}$ de um novo nó v_i se conectar a um nó existente v_j é:

$$p_{i,j} = \frac{d_j}{\sum_{u=1}^{n} d_u}.$$

Em Leskovec et al. (2008), o modelo BA foi extensivamente validado a partir de quatro coleções de dados reais: Flickr, Delicious, Yahoo! Answers e Linkedin. Foi verificado que, mesmo comparado com variações mais sofisticadas, o modelo BA é aquele que melhor explica as bases de dados analisadas. No entanto, o modelo BA leva em consideração o conhecimento global da rede por parte dos nós, o que não é realista para a maior parte das redes. Também em Leskovec et al. (2008) e em Vázquez (2003), os autores

propõem uma versão local para o modelo BA, em que o processo de criação das arestas não depende do conhecimento global da rede. As arestas são criadas com o propósito de fechar triângulos, considerando unicamente da vizinhança do nó que ingressara na rede.

Além disso, há outras características inerentes a esse tipo de rede ou a variações dela que o modelo falha em capturar. Devido a isso, outros modelos para geração de RLEs foram propostos na literatura. Em Chen & Shi (2004), foram propostos duas variações do modelo BA que consideram a realocação e a deleção de arestas da rede. Em Holme & Kim (2002), os autores definiram um passo adicional ao modelo BA, em que cada aresta criada pelo modelo BA gera uma segunda aresta adicional que fecha um triangulo. Esse modelo é interessante porque cria uma rede com características de uma RLE e de uma rede *small-world*. Em Leskovec et al. (2007) é proposto o modelo *forest fire*, que além das características previamente descritas, também é capaz de reproduzir duas novas observações temporais feitas a partir de análises de redes reais: densificação e diminuição do diâmetro. Densificação estipula que o número de arestas cresce super-linearmente em relação ao crescimento do número de nós, ao mesmo tempo em que o diâmetro da rede diminui com o tempo.

Outro cenário interessante que foi investigado é aquele em que as redes possuem pesos nas arestas. Em McGlohon et al. (2008), os autores propuseram o modelo *butterfly*, em que foi verificado o efeito de fortificação da rede com o tempo, que estipula que a relação entre o número de arestas de um grafo e o peso total das mesmas é super linear, seguindo uma lei de potência. Tal efeito também é válido para explicar a relação entre o grau de um nó e o peso total das suas arestas. Em Du et al. (2009), os autores propuseram o modelo *PAC*, que reproduz os padrões dos pesos dos triângulos em grafo, mostrando que as distribuições do peso maior, do intermediário e do menor em um triângulo seguem leis de potência em que os expoentes não variam com o tempo. Por fim, o modelo RTG (Akoglu & Faloutsos 2009) é um modelo linear de quatro parâmetros que serve como uma interessante alternativa para gerar RLEs com todas as características descritas nesta seção. Para mais informações sobre modelos geradores de RLEs, consulte Mitzenmacher (2004), Chakrabarti & Faloutsos (2006), Zhou & Lipowsky (2005).

1.4.3.2. Propriedades em Redes de Computadores

Uma característica imediata que pode-se imaginar de uma rede de computadores que tenha a topologia de uma RLE é a alta tolerância a falhas. Como foi mostrado em Albert et al. (2000), a grande maioria dos nós possui baixo grau, a remoção de nós aleatórios da rede, mesmo que em grade escala, mantém a rede conectada. No entanto, esse tipo de rede é altamente vulnerável a ataques, ou seja, a remoção dos poucos *hubs* da rede a desconecta. Além disso, em Motter et al. (2002) é mostrado que redes com topologia RLE são mais robustas a ataques feitos a arestas de longo alcance, isto é, arestas que conectam nós que de outra maneira seriam separados por uma longa distancia em saltos, que redes com topologias aleatórias e com conectividade segundo a distribuição de Poisson. Além disso, foi mostrado que RLEs são mais vulneráveis a ataques feitos a arestas de curto alcance que de longo alcance, pois tais arestas são responsáveis por um maior tráfego, uma vez que conectam dois nós de alto grau e, consequentemente, possuem um alto valor de

betweeness.

Outra característica interessante de redes de computadores com topologia de RLEs é quanto à busca por nós ou por informações. Em Adamic et al. (2001), os autores mostraram que um simples algoritmo local de navegação, que escolhe como próximo nó na busca o nó de maior grau, tem sucesso na busca a um custo sublinear em relação ao tamanho da rede. Isso permite que nós remotos de redes informais, como sistemas P2P por exemplo, sejam alcançados através de arestas locais e, consequentemente, permite que pessoas e sistemas de comunicação consigam distribuir informação e acessar os recursos que desejam. Nessa direção, Sarshar & Roychowdhury (2004) propuseram um protocolo que permite a formação de topologias livre de escala mesmo em redes *ad-hoc* altamente dinâmicas, como redes P2P em que nós frequentemente se conectam e desconectam da rede. Guclu et al. (2008) propuseram um modelo de busca para um cenário em que os nós desse tipo de rede possuem informação limitada, ou seja, em que há um limite para o número de vizinhos que um nó pode ter, o que faz com que não haja *hub* majoritário na rede.

1.4.4. Comunidades

Algumas redes, em especial as redes sociais, biológicas e tecnológicas, como a Internet, tendem a apresentar uma estrutura modular (Girvan & Newman 2002). Tal estrutura modular é caracterizada pela existência de um conjunto ou comunidade de vértices, onde a maioria das conexões se dá entre vértices pertencentes a uma mesma comunidade, enquanto que conexões entre vértices pertencentes a comunidades diferentes são menos frequentes. A formação de tais comunidades pode ser dar por algum tipo interesse, idade, trabalho, páginas mais frequentemente visitadas e assim por diante.

Girvan & Newman (2002) propuseram um modelo para a geração de redes com tal característica, onde esse modelo pode ser visto como uma generalização do modelo para a geração de grafos aleatórios. Nesse modelo, a primeira etapa consiste na classificação de m vértices em c comunidades. Os passos seguintes consistem na seleção de dois vértices e adição de uma aresta com probabilidade p_{in}, caso ambos os vértices pertençam a mesma comunidade, ou p_{out} , caso os vértices pertençam a comunidades diferentes. É importante observar que a escolha dos valores de p_{in} e p_{out} deve ser feito de forma cuidadosa, de forma que a distinção entre comunidades seja o mais nítida possível.

Por outro lado, o método tradicional para extração de estruturas de comunidade de uma rede é conhecido como *análise de agrupamentos* ou *agrupamento hierárquico*. Nesse método, é atribuído um fator conhecido como *connection strength* a pares de vértices na rede de interesse. Em seguida, iniciando com n vértices que não possuem qualquer aresta entre si, adiciona-se arestas de acordo com a ordem decrescente do fator *connection strength* entre vértices. Pode-se interromper esse processo em qualquer momento de forma a examinar a estrutura do componente formado pela adição de arestas até então. Tais componentes são tidos como as comunidades ou agrupamentos. Quando todas as arestas forem adicionadas, todos os vértices estarão conectados a todos os outros e existirá apenas uma comunidade. A escolha para os valores do fator *connection strength* pode levar em conta questões como distância entre vértices, fluxo de dados, entre outros.

Uma questão interessante relacionada as redes que apresentam estruturas de co-

munidades é: dado um vértice pertencente a uma rede, é possível identificar a qual comunidade o mesmo pertence. Algoritmos que responda a tal questão pode ser de fundamental importância para alguns tipos de redes, como por exemplo, a *World Wide Web*.

1.4.5. Geográficas

O modelo de redes geográficas (ver Gastner & Newman 2006) incorpora um elemento a mais que os outros modelos apresentados não consideram: o posicionamento dos vértices no espaço. Desta maneira, a probabilidade de uma aresta existir depende do posicionamento de cada um dos vértices, e especialmente a distância dentre eles.

O modelo mais simples de rede geográfica é o GRG (*Geometric Random Graph*). Neste modelo, os vértices são distribuídos aleatoriamente em um espaço Ω. A conectividade entre dois nós quaisquer u e v é dada como função da distância Euclideana $\delta(u,v)$ e uma constante que determina o limiar de comunicação r,

$$(u,v) \in E(G) \iff \delta(u,v) \leq r,$$

onde $E(G)$ é o conjunto de arestas do grafo não direcionado G. O modelo GRG segue uma regra rígida baseada na distância Euclideana, o que é útil para modelar comunicações sem fios em ambientes ideais.

Outros modelos são usualmente empregados para modelar a conectividade como uma função decrescente em relação à distância. Por exemplo, instâncias de redes geográficas podem ser geradas seguindo a seguinte função de probabilidade:

$$\Pr((u,v) \in E(G)) \sim e^{-\varepsilon\delta(u,v)},$$

onde ε fixa o a escala de comprimento das arestas no grafo.

Outras maneiras de modelar o posicionamento dos vértices no espaço podem ser utilizadas. Em particular, uma metodologia que facilita muito a modelagem é expressar o posicionamento dos vértices por meio de processos pontuais estocásticos (ver seção 1.5.1 para maior detalhamento sobre processos pontuais).

1.5. Redes complexas na modelagem de redes de computadores

Nesta seção trataremos, inicialmente, dos métodos estatísticos necessários para a aplicação da teoria das redes complexas e em seguida discutiremos diversas aplicações

1.5.1. Métodos estatísticos

Consideremos uma rede de sensores sem fios (ver seção 1.5.2.4), onde os nós sensores estão posicionados de maneira aleatória numa área de interesse. Neste caso temos redes espaciais, também conhecidas por redes geográficas. Agora, do ponto de vista da modelagem, podemos nos fazer as seguintes perguntas:

- Como podemos modelar o posicionamento?
- O que significa precisamente “posicionados de maneira aleatória”?
- Como podemos determinar se há conectividade entre dois nós quaisquer da rede?

- Há algum modelo de rede complexa capaz de capturar a estrutura desta rede?

Podemos utilizar uma modelagem no espaço Euclideano bi- ou tri-dimensional onde o posicionamento de um sensor qualquer é representado pela tupla (x,y) ou (x,y,z), respectivamente, onde x, y e z são valores numéricos. O tipo de valor numérico das coordenadas Euclideanas depende da nossa modelagem. Para simplificar, de agora em diante, vamos considerar a modelagem bi-dimensional numa região $W = [0,\ell]^2 \subset \mathbb{R}^2$, portanto consideramos o posicionamento de cada sensor como sendo uma tupla (x,y) com $0 \leq x \leq \ell$ e $0 \leq y \leq \ell$.

Para modelar um conjunto de sensores distribuídos aleatoriamente seguindo alguma lei de probabilidade, utilizamos processos pontuais espaciais, que são distribuições de probabilidade capazes de descrever a localização espacial de pontos (ver Baddeley 2007, Berthelsen & Møller 2002, Møller & Waagepetersen 2007, e as referências aí citadas). O processo pontual mais utilizado na área de redes de sensores é o URP (*Uniform Random Placement*)

O URP é conhecido na literatura de processos pontuais como *Processo Pontual de Poisson*. Ele é tido como referência de aleatoriedade, para fins de comparação com outros processos pontuais. A forma construtiva de definir o URP com intensidade $\eta > 0$ sobre a região W consiste das seguintes etapas:

1. Observar n, ocorrência da variável aleatória N com distribuição de Poisson e parâmetro $\lambda = \eta\ell^2$, isto é, $\Pr(N = k) = e^{-\lambda}\lambda^k/k!$.

2. Observar $2n$ ocorrências de variáveis aleatórias independentes e identicamente distribuídas no intervalo $[0,\ell]$, por exemplo, $(x_1,y_1,\ldots,x_n,y_n)$.

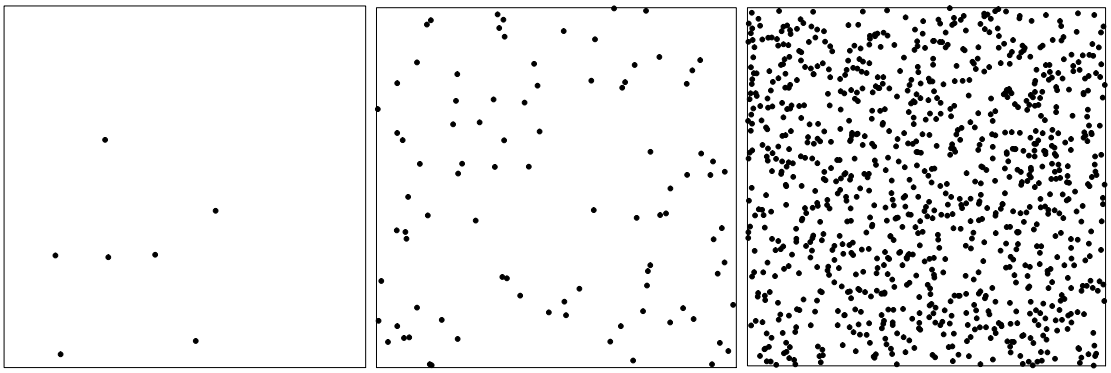
3. Posicionar n pontos na região W nas coordenadas $(x_1,y_1),\ldots,(x_n,y_n)$.

Este processo é considerado como referência pois ele possui (de fato, ele é caracterizado por) as seguintes propriedades:

1. Sejam $A_1, A_2 \subset W$ disjuntos, então $\Pr(N_{A_1} = k_1, N_{A_2} = k_2) = \Pr(N_{A_1} = k_1)\Pr(N_{A_2} = k_2)$, isto é, há independência entre as variáveis aleatórias que descrevem o número de pontos em regiões disjuntas.

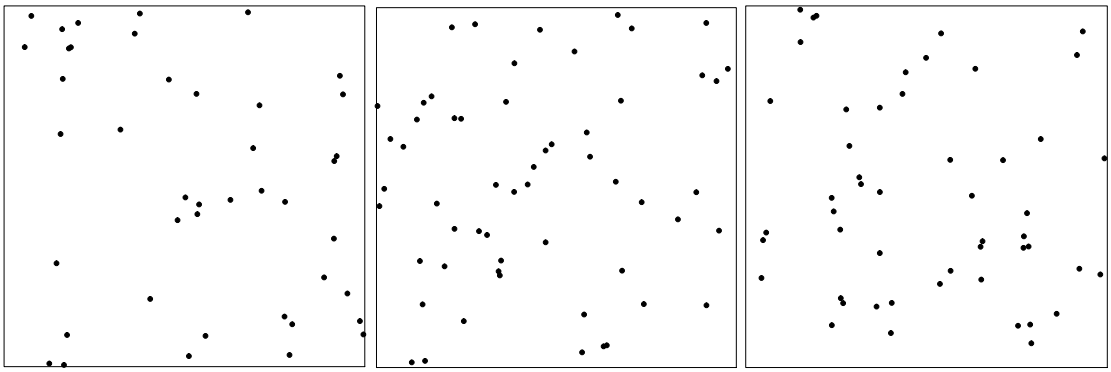
2. A probabilidade de observar $k \geq 0$ pontos em uma região $A \subset W$ segue uma distribuição de Poisson com média $\lambda_A = \eta\mu(A)$, onde $\mu(A)$ é a área do conjunto A, isto é, $\Pr(N_A = k) = e^{-\lambda_A}\lambda_A^k/k!$, onde N_A é o número de pontos no conjunto A.

Tal como dito anteriormente, o URP é o modelo mais frequentemente encontrado na modelagem de deposição de nós sensores. Há, contudo, outros modelos de processos pontuais aptos a descrever situações onde os pontos tendem a se aglomerar (processos atrativos) ou a se repelir (processos repulsivos). Frery et al. (2008) propoem um modelo composto que, através de um único parâmetro, exibe os três comportamentos. No que



(a) $\eta = 10$ (b) $\eta = 100$ (c) $\eta = 1000$

Figura 1.6. Três eventos do modelo URP definido sobre a mesma janela $W = [0,1]^2$ com intensidade η variável



(a) $n = 42$ (b) $n = 57$ (c) $n = 51$

Figura 1.7. Três eventos do modelo URP com intensidade $\eta = 50$ definido sobre a janela $W = [0,1]^2$

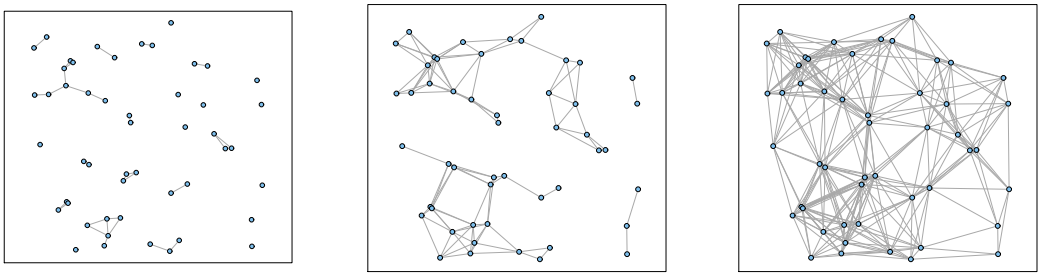
segue, empregaremos apenas o URP. A figura 1.6 ilustra três casos de URP com intensidade variável. A figura 1.7 ilustra três eventos do mesmo modelo de URP, onde constatamos que o número de sensores varia de evento para evento, mesmo sendo a intensidade constante.

Uma vez depostos os n sensores na região W segundo o URP, precisamos estipular o modelo de conectividade induzido pela localização espacial dos nós. Por simplicidade pode-se adotar o modelo UDG (*Unit Disk Graph*). Outros modelos mais realistas podem ser encontrados em Tselishchev et al. (2010). O modelo UDG consiste em considerar que os nós u e v, localizados nos pontos (x_u, y_u) e (x_v, y_v) respectivamente, são vizinhos no grafo de conectividade, isto é, se comunicam, se a distância Euclidiana entre eles é no máximo δ_c, isto é

$$(u,v) \in E \iff (x_u - x_v)^2 + (y_u - y_v)^2 \leq \delta_c^2.$$

O modelo de comunicação induzido pelo UDG é bilateral, isto é, se u se comunica com v, então necessariamente v se comunica com u. Há outros modelos de conectividade, inclusive unilaterais.

Com os dois ingredientes descritos acima, isto é, com o modelo de deposição URP e o modelo de comunicação UDG, já podemos simular redes de sensores sem fios. Basta, para tanto, estipular a área de interesse W através do lado ℓ, a intensidade do processo η e o raio de comunicação δ_c. A tripla (ℓ, η, δ_c) estipula ou indexa um modelo da classe dos modelos geográficos, nos quais a topologia não depende dos dados mas apenas da localização. Esta classe de modelos é discutida em mais detalhes na seção 1.4.5. A figura 1.8 ilustra três grafos de comunicação induzidos por três raios de comunicação diferentes sobre o mesmo conjunto de $n = 50$ pontos.



(a) $\delta_c = 0.1$ (b) $\delta_c = 0.2$ (c) $\delta_c = 0.35$

Figura 1.8. Três eventos do modelo URP com intensidade $\eta = 50$ definido sobre a janela $W = [0,1]^2$

Tal como construído até o momento, um grafo de conectividade como os mostrados na figura 1.8 é um evento de um processo aleatório caracterizado pela tripla (ℓ, η, δ_c). Medições efetuadas sobre um desses eventos serão eventos de variáveis aleatórias. Por exemplo, se simularmos uma sequência de grafos independentes $g_1, g_2, \ldots, g_N$ cuja distribuição é caracterizada pela tripla (ℓ, η, δ_c), e em cada um deles medirmos o grau médio do grafo $k_m(g_i)$, teremos uma sequência de eventos $k_m(g_1), k_m(g_2), \ldots, k_m(g_N)$. Essa sequência de valores poderá ser analisada como sendo eventos de variáveis aleatórias independentes, também caracterizadas pela tripla (ℓ, η, δ_c). Em outras palavras, neste ponto do texto dispomos de um modelo estocástico para a variável aleatória G, definida sobre o espaço amostral genérico Ω com valores no conjunto de grafos de conectividade $\mathscr{C}$, i.e., $G\colon \Omega \to \mathscr{C}$. Esse modelo é indexado pela tripla (ℓ, η, δ_c), isto é, a probabilidade de observar um grafo particular $\Pr(G = g)$ depende apenas de (ℓ, η, δ_c). A seguir definiremos variáveis aleatórias reais sobre esses grafos, e procederemos a analisá-las. Sem perda de generalidade, iremos considerar sempre o mesmo suporte do processo pontual, W, fixando $\ell = 1$ e, portanto, indexando o modelo apenas pela dupla (η, δ_c).

Considere o grafo aleatório $G = (V, E)$, onde V é o conjunto de nós (produzido por um URP de intensidade η sobre a janela quadrada de lado unitário) e V é o conjunto de arcos (induzido pelo raio de conectividade δ_c).

A seguir iremos montar um experimento Monte Carlo (Cipra 2000, Dongarra & Sullivan 2000, Robert & Casella 2000) para verificar a hipótese "não sei nada" em uma boa diversidade de situações. Idealmente, a nossa hipótese de trabalho deveria ser verificada analiticamente para todos os valores de (η, δ_c) e, com isso, acabaria a dúvida.

A técnica Monte Carlo é uma ferramenta de aplicabilidade geral, que só deve ser usada quando outras abordagens falham ou seriam muito demoradas para dar uma resposta aceitável.

O primeiro passo consiste em discretizar o conjunto de todas as situações de interesse formando, assim, o conjunto $\Theta = [\eta_1, \ldots, \eta_{M_1}] \times [\delta_{c1}, \ldots, \delta_{cM_2}]$. O tamanho desse conjunto, isto é, o valor $M_1 M_2$, e a abrangência dos intervalos deverão ser tão grandes quanto possível, mas sem comprometer a execução da experiência pelo tempo demandado.

O segundo passo consiste em montar os algoritmos de simulação, isto é, as técnicas que irão produzir os eventos do grafo aleatório G para cada $\theta \in \Theta$. Na nossa experiência, recomendamos empregar funções disponíveis através dos pacotes `spatstat` (para simulação de processos pontuais) e `igraph` (para computação com grafos) da plataforma R (R Development Core Team 2009).

A estrutura do nosso experimento Monte Carlo é mostrada no algoritmo 1.

Algoritmo 1 Estrutura do experimento Monte Carlo para análise de grau médio, máximo *betweenness*, diâmetro e vulnerabilidade de grafos aleatórios

```
1: for cada η ∈ [η_1, ..., η_{M_1}] do
2:    for cada δ_c ∈ [δ_{c1}, ..., δ_{cM_2}] do
3:       for cada r ∈ {1, ..., N_R} do
4:          Gerar o evento g_r do grafo aleatório G(η, δ_c)
5:          Calcular e armazenar (k_m, B_max, d, V)(g_r)
6:       end for
7:    end for
8: end for
9: Analisar resultados e procurar padrões
```

Para poder implementar o algoritmo 1 precisamos estipular o número de replicações N_R. Em situações relativamentes simples podemos lançar mão da regra de bolso $N_R = 30$ como mínimo ou $N_R \geq 100$ para termos resultados mais confiáveis. Essa regra de bolso se baseia em uma confiança um pouco cega na validade do teorema central do limite. Para fazermos uma análise quantitativa mais cuidadosa, podemos seguir a apresentação feita por Díaz-Emparanza (1996).

Vamos a seguir considerar a situação de querermos estimar uma proporção p, desconhecida, através da repetição de N experimentos dicotômicos independentes e identicamente distribuídos. Pelo fato de serem dicotômicos, o resultado possível é ou "Verdadeiro" ou "Falso", e não há nenhuma outra possibilidade. Atribuamos o valor 1 ao primeiro resultado possível, e 0 ao segundo. Podemos estimar p através de

$$\widehat{p} = \frac{1}{N} \sum_{i=1}^{N} X_i, \tag{1}$$

onde X_i é a variável aleatória que modela o experimento número i. É importante frisar que $\widehat{p}$ é uma variável aleatória, pois é o resultado de transformar as variáveis aleatórias

$X_1, \ldots, X_N$. Dizemos que $\widehat{p}$ é um *estimador* de p; uma vez realizada a experiência e observado um valor particular, por exemplo, se fizermos o experimento e observarmos dezenove sucessos em cinquenta tentativas, não temos mais um estimador mas sim uma *estimativa*: teremos $\widehat{p}(\omega) = 19/50$, onde ω denota um ponto do espaço amostral Ω.

O que acontecerá se repetirmos o experimento? Tipicamente teremos outra estimativa. Se fosse possível repetir infinitas vezes o experimento, ou percorrer todo Ω, poderíamos ter uma idéia da variabilidade do nosso estimador $\widehat{p}$, mas essa abordagem não é conveniente e muito raramente é viável.

Ao invés de procedermos por enumeração sobre todo o espaço amostral Ω, podemos obter um bom conhecimento da qualidade do procedimento de estimação (ou inferência) analisando as propriedades de $\widehat{p}$, variável aleatória, que decorrem da sua definição dada na equação (1).

Uma preocupação subjacente a qualquer experiência empírica é a sua *precisão*. No nosso contexto, a precisão de um estimador está associada ao seu viés, que é definido como a diferença entre a esperança do estimador e o valor que ele estima, isto é

$$B(\widehat{p}, p) = \mathrm{E}(\widehat{p}) - p, \tag{2}$$

onde B é pelo termo em Inglês *bias*, e E denota a esperança matemática. O viés de um estimador tipicamente depende do tamanho da amostra N e do verdadeiro valor do parâmetro. Um estimador que possui viés nulo para qualquer N e qualquer valor do parâmetro, é denominado *não viesado* (*unbiased* em Inglês).

Embora o viés seja uma propriedade muito importante, ao ponto de ser um ativo objeto de pesquisa na estatística (ver, por exemplo, Cox & Snell 1968, e a miríade de artigos que seguiram a esse trabalho pioneiro), ele não fornece uma idéia da variabilidade do estimador. Essa variabilidade é um elemento fundamental, junto com o viés, para poder estipular comparações entre estimadores. Mediremos a variabilidade de um estimador através da acurácia; dizemos que o estimador $\widehat{p}$ tem acurácia a ao nível $1-\alpha$, com $\alpha \in (0,1)$ se

$$\Pr(|\widehat{p} - p| < a) \geq 1 - \alpha. \tag{3}$$

É importante interpretar os elementos desta desigualdade.

Voltemos à nossa experiência imaginária de obter infinitas observações independentes da variável aleatória $\widehat{p}$, com p e N fixos. Algumas estimativas ficarão bem próximas de p, outras nem tanto e umas poucas irão ficar bem distantes do verdadeiro valor do parâmetro. Fixando uma tolerância a (a nossa acurácia), e seguindo uma abordagem frequentista, podemos ter uma idéia de quão frequentemente o estimador $\widehat{p}$ se afasta do verdadeiro valor p menos do que a. Conhecendo a distribuição do estimador, podemos inclusive calcular a probabilidade disso ocorrer; temos então a equação (3). Intuitivamente, se aumentarmos o tamanho da amostra N, o nosso estimador ficará cada vez mais confinado ao intervalo em volta do verdadeiro valor e, com isso, a probabilidade da equação (3) aumentará com N. Valores típicos dessa probabilidade são $90/100$, $95/100$, $99/100$ e $999/1000$, que correspondem a $\alpha = 1/10, 5/100, 1/100$ e $1/1000$, respectivamente.

Podemos pensar de forma dual, isto é, ao invés de aumentar o tamanho da amostra N até alcançarmos a probabilidade (ou α) alvo, estipular a probabilidade (ou o valor de

α) e calcular o tamanho mínimo da amostra que nos dará a acurácia a desejada. Como exemplo, suponhamos que desejamos $a = 1/100$ e $\alpha = 5/100$, isto é, desejamos calcular o tamanho da amostra N tal que

$$\Pr(|\widehat{p} - p| < 10^{-2}) \geq 95/100 \tag{4}$$

para todo valor de $p \in (0,1)$. Reescrevendo a equação (4) temos

$$\Pr(-10^{-2} < \widehat{p} - p < 10^{-2}) \geq 95/100. \tag{5}$$

Lembrando que a variância de $\widehat{p}$ é $p(1-p)/N$,

$$\Pr\left(-\frac{10^{-2}}{\sqrt{\frac{p(1-p)}{N}}} < \frac{\widehat{p} - p}{\sqrt{\frac{p(1-p)}{N}}} < \frac{10^{-2}}{\sqrt{\frac{p(1-p)}{N}}} \right) \geq 95/100. \tag{6}$$

O Teorema Central do Limite (recomendamos ver a formulação de James 1981) afirma que, sob condições bastante gerais, a soma de variáveis aleatórias convenientemente padronizada é uma variável aleatória cuja distribuição pode ser aproximada por uma lei gaussiana. Em particular, se $X_1, \ldots, X_N$ são variáveis aleatórias independentes e identicamente distribuídas com esperança μ e variância σ^2, então

$$\Pr\left(\frac{\frac{1}{N}\sum_{i=1}^{N} X_i - \mu}{\sigma} \in I \right) \rightarrow \int_I \frac{1}{\sqrt{2\pi}} \exp\{-t^2/2\} dt, \tag{7}$$

quando $N \rightarrow \infty$, para qualquer intervalo I. A integral do lado direito encontra-se tabulada na maioria dos livros de estatística, e está implementada em virtualmente todas as plataformas de software de análise de dados.

Se for possível considerar válido o Teorema Central do Limite para a variável aleatória

$$Z = \frac{\widehat{p} - p}{\sqrt{\frac{p(1-p)}{N}}},$$

então é imediato que $\Pr(-1,96 \leq Z \leq 1,96) \approx 95/100$ e, com isso,

$$1,96 = \frac{10^{-2}}{\sqrt{\frac{p(1-p)}{N}}},$$

e daí

$$N = (1,96)^2 10^4 p(1-p). \tag{8}$$

A equação (8) não resolve o nosso problema pois, segundo ela, para determinar o tamanho da amostra necessária para estimar p ao nível de confiança $95/100$ com precisão 10^{-3} precisamos do valor de p. A saída para este aparente dilema circular advém do uso de simulação: podemos substituir p na equação (8) por uma primeira estimativa $\widetilde{p} = M^{-1}\sum_{j=1}^{M} X_j$ baseada em M replicações, com M um número aceitável de replicações. Por "aceitável", entenda-se o maior número possível de replicações cujo tempo de obtenção for compatível com o estudo.

Se na nossa simulação exploratória determinarmos $\widetilde{p} = 1/10$, então o experimento Monte Carlo que precisamos realizar requer $N = (1,96)^2 10^4 (9/10)/10 \approx 3457$ replicações.

Voltando à equação (8), vamos escrevê-la com os valores teóricos ao invés dos numéricos empregados para criar o exemplo:

$$N = \left(\frac{t_{\alpha/2}}{a}\right)^2 p(1-p). \tag{9}$$

A equação (9) fornece uma relação útil entre o número mínimo de replicações N necessário para ter precisão a ao nível de confiança $1-\alpha$ na estimação da proporção p, onde t_ν é o quantil de ordem ν da distribuição gaussiana padrão. Ela permite calcular o número mínimo de replicações em função de a e de α, ou conhecer a precisão em função da significância e do número de replicações, isto é, o intervalo de confiança.

A noção de "intervalo de confiança" é frequentemente mal empregada. É comum ler que "a estimativa estará em um intervalo de tamanho a ao redor do verdadeiro valor com probabilidade $1-\alpha$", que é uma interpretação errada. Uma vez observada a estimativa, ela não é mais uma variável aleatória e, portanto, qualquer câlculo de probabilidade ao seu respeito não faz sentido. A interpretação correta (do ponto de vista frequentista) é "se repetirmos infinitas vezes a experiência sob as mesmas condições, em uma proporção $1-\alpha$ desses ensaios a estimativa estará dentro do intervalo de tamanho a centrado no verdadeiro valor p". A diferença é sutil, porém contundente.

O procedimento descrito acima pode ser facilmente adaptado para determinar o número mínimo de replicações necessário para alcançar uma certa precisão com um certo nível de confiança. Na maioria dos casos, será necessário o conhecimento da distribuição do estimador, bem como a realização de pelo menos uma experiência de simulação exploratória. Mais detalhes podem ser vistos em Bustos & Frery (1992).

A seguir, daremos algumas dicas sobre o significado do passo 9 do algoritmo 1, isto é, sobre o que é *analisar* dados de um ensaio Monte Carlo. Em primeiro lugar, os dados de um ensaio Monte Carlo devem ser tratados como dados de qualquer experiência empírica, isto é, devem ser convenientemente catalogados e organizados.

Toda análise de dados é formada por operações que podem ser categorizadas em (i) análise qualitativa, e (ii) análise quantitativa. A análise qualitativa costuma ser exploratória, enquanto a quantitativa é mais confirmatória. A primeira procura hipóteses a serem verificadas ou descartadas com a segunda. Ambas podem ser feitas com ferramentas gráficas ou quantitativas. Não se trata de atividades lineares, no sentido de uma preceder a outra; são operações que se realizam conforme o analista vai conhecendo os dados à procura de informações relevantes. Um bom analista dispõe de um amplo ferramental, e de experiência para guiar as atividades.

Everitt (2005) apresenta uma introdução às principais técnicas de análise de dados multivariados, tanto para análise qualitativa quanto quantitativa, utilizando R. Para aplicar qualquer um destes procedimentos, como mencionado anteriormente, é importante manter os dados organizados corretamente para conseguir aplicar as diferentes operações de análise de dados sem a necessidade de trocar de formatação. A formato de dados estatísticos usual é o *data frame*, e ele tem a forma de uma matriz, onde cada fila representa uma

instância do experimento, e cada coluna pode ser um *fator*, que caracteriza a instância "simulada", ou uma medida de interesse registrada para essa instância. Uma boa referência sobre a organização dos dados (e sobre análise de dados em geral) é Chambers (2008). Recomendamos aos leitores ler especificamente a seção 5 do capítulo 6, que trata sobre este tópico em particular.

1.5.2. Aplicações

A seguir, apresentamos alguns resultados obtidos pela comunidade de pesquisadores que vêm empregando a teoria de redes complexas em diferentes áreas de pesquisa relacionadas com redes de computadores. Cada uma das seções seguintes é dedicada a tipos de redes diferentes. A classificação escolhida é baseada na modelagem e propósitos destas redes. A seção 1.5.2.1 apresenta os resultados relacionados com a Internet. A aplicação de teoria de redes complexas em redes *peer-to-peer* são apresentadas na seção 1.5.2.2. Na seção 1.5.2.3 consideramos aplicações no contexto de redes ad-hoc. Finalmente, a seção 1.5.2.4 considera aplicações em redes de sensores sem fios.

Cada uma das seções mencionadas acima considera a modelagem dessas redes do ponto de vista de redes complexas. A figura 1.9 apresenta algumas das áreas onde a teoria de redes complexas já foi empregada com êxito.

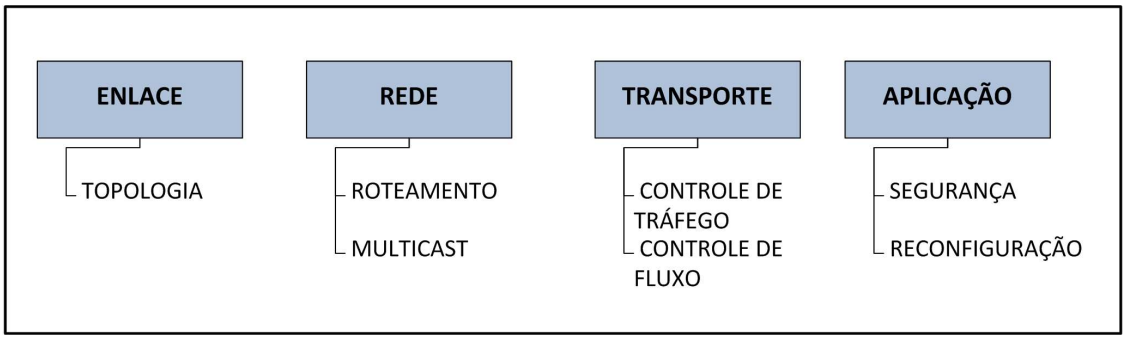


Figura 1.9. Taxonomia de áreas de aplicação de redes complexas

1.5.2.1. Internet

Faloutsos et al. (1999) apresentam o primeiro trabalho de modelagem, do ponto de vista topológico, da Internet. Como os autores citam neste trabalho, a compreensão da topologia da Internet traz benefícios, dentre eles:

- conhecendo melhor a topologia, seremos capazes de projetar protocolos mais eficientes que tirem proveito das propriedades topológicas presentes na Internet;
- é possível obter modelos de redes mais precisos para efetuar simulações, e;
- podemos derivar estimadores de parâmetros topológicos para a análise de protocolos, assim como também obter predições da evolução da topologia da Internet no futuro.

Siganos et al. (2003) apresentam um estudo da topologia da Internet ao nível de sistemas autônomos, propondo modelos de leis de potências da forma $y \propto x^a$ nesta hierarquia, onde a é uma constante e x e y são medidas de interesse. Os autores coletaram informações das tabelas de roteamento BGP quase diariamente por um período de 5 anos (desde 1999 até 2002), e em cada instância do tempo foram validados os modelos de leis de potência para diferentes medidas topológicas. Em particular foram consideradas as seguintes:

- D_d: a função de distribuição acumulada complementar de um grau, que é a porcentagem de nós que possuim grau maior ou igual a d;
- r_v: a ordem de um nó, v, é o índice na ordem decrescente de graus;
- $P(h)$: o "número de pares" de nós, é o número total de pares ordenados de nós que estão como máximo a h *hops* de distância;
- $NN(h)$: o número médio de nós em uma vizinhança de h *hops*;
- λ: o autovetor do grafo de comunicações;
- i: o autovalor de ordem i do autovetor λ ordenado.

Baseado nestas medidas, as seguintes leis de potência foram validadas:

- expoente de *rank*: dado um grafo, o grau d_v de um nó v é proporcional ao *rank* r_v à potência de uma constante $\mathscr{R}$, isto é, $d_v \propto {r_v}^{\mathscr{R}}$;
- expoente de grau: dado um grafo, a função de distribuição acumulada complementar de um grau D_d de d é proporcional ao grau à potência de uma constante $\mathscr{D}$, isto é, $D_d \propto d^{\mathscr{D}}$;
- auto-expoente: dado um grafo, os autovalores λ_i são proporcionais ao ordem i à potência de uma constante $\mathscr{E}$, isto é, $\lambda_i \propto i^{\mathscr{E}}$.

O trabalho de Siganos et al. (2003) apresenta um caracterização da topologia de sistemas autônomos da Internet através de leis de potência. Apesar de que os autores consideraram o fator temporal sobre estas leis, elas não são suficientes para representar todas as características topológicas inter-domínio. As leis apresentadas aqui (e outras de este e trabalhos posteriores) são uma condição necessária mas não suficiente para obtermos uma modelagem que permita conhecer a evolução topológica da Internet.

1.5.2.2. Redes *peer-to-peer*

No mundo real é possível observar a existência de vários tipos de redes sociais, como redes de conhecimento e colaboração, onde estudos realizados em redes sociais têm sido aplicados em outros tipos de redes, como as tecnológicas e as biológicas. Isso decorre do fato de que a grande maioria das redes sociais apresentam propriedades desejáveis para

esses outros tipos de redes, como por exemplo, o efeito *small world*, *scale free* e a existência de comunidades. Muitos modelos interessantes foram propostos de forma a obter redes com determinadas propriedades em uma escala global. No entanto, em redes reais, todo o controle e o gerenciamento sobre a estrutura da rede está implícita sobre as escolhas locais realizadas pelos pares de forma distribuída e não-supervisionada. Cada par escolhe seu vizinho e liga-se ao mesmo obedecendo apenas seus interesses pessoais, sem levar em consideração a formação de uma rede com uma estrutura ou propriedade desejada. Diante disso, no trabalho proposto por Carchiolo et al. (2008) é apresentando um modelo para o crescimento e evolução de redes *peer-to-peer* inspiradas na dinamicidade de redes sociais, modelo conhecido como *PROSA*. Esse modelo é capaz de gerar redes com características desejáveis, tais como *small world* e estruturas de comunidade, além de manter os mecanismos de comunicação entre pares e gerenciamento de rede simples e intuitivos. Os mecanismos utilizados no gerenciamento de ligações e roteamento de consultas induzem a formação de uma forte estrutura de comunidades, a qual pode explicar o porque a busca e recuperação de recursos no *PROSA* é tão rápida e eficiente.

A principal consideração realizada pelo modelo *PROSA* é que as propriedades globais de muitas redes sociais devem-se ao comportamento local dos pares. A principal hipótese é que é possível obter características globais semelhantes à uma rede social real em uma rede P2P se o comportamento local dos pares for similar ao comportamento de pessoas em uma rede social real. Por essa razão, *PROSA* é fortemente inspirado pelas redes de relacionamentos entre pessoas. Dessa forma, a rede P2P formada pelo modelo *PROSA* apresenta características como caminho mínimo médio baixo entre os pares da rede, alto coeficiente de agrupamento e a presença de fortes estruturas de comunidade. Os pares e as ligações no *PROSA* evoluem como pessoas e relacionamentos entre pessoas em uma rede social, ou seja, cada par liga-se a um certo número de outros pares, repassam buscas para tais pares e ocasionalmente tomam conhecimento de novos pares e estabelece ligações com eles. Pares enviam mensagens somente para os seus parentes e vizinhos; mensagens para os pares desconhecidos são repassadas através de rotas que provavelmente irão encontrá-los, olhando apenas para as conexões e conhecimentos locais. O resultado é uma rede com gerenciamento totalmente descentralizado e distribuído, a qual apresenta muitas propriedades estruturais de redes sociais locais.

Já o protocolo *SWOP* proposto em Hui et al. (2006) utiliza-se das propriedades presentes nas redes do tipo *small world* para a criação e manutenção de uma rede estruturada *peer-to-peer*. Nesse trabalho, duas questões fundamentais aos sistemas *P2P* são trabalhadas:

- como melhorar o desempenho na busca por objetos dentro da rede;
- como uma rede *P2P* pode manipular a demanda elevada por objetos populares e altamente dinâmicos.

Para solucionar a primeira questão, os autores propuseram a criação de uma rede *P2P* com as propriedades de uma rede *small world*. Ou seja, ao se utilizar tal modelo na criação de uma rede *P2P* a rede passaria a apresentar um baixo caminho mínimo médio entre nós aleatórios, resultando na melhoria da busca por objetos dentro da rede.

Já a segunda questão é de fundamental importância, pois alguns objetos podem ser extremamente populares e requisições por tais objetos podem ser recebidas dentro de uma intervalo de tempo relativamente pequeno. Esse tipo de tráfego pode sobrecarregar o nó que possuir tal objeto e consequentemente muitos usuários podem não ter acesso a esse objeto. Na tentativa de solucionar tal problema, os autores tiraram vantagem do alto grau de agrupamento existente nas redes do tipo *small world* de forma que os nós dentro de uma vizinhança podem se auto-organizar e replicar o objeto altamente popular entre eles e dessa forma tornar o sistema fortemente robusto em situações de tráfego intenso.

1.5.2.3. Redes ad-hoc

Uma rede de computadores *ad hoc* é uma rede que não precisa de infraestrutura especial para os nós estabelecerem comunicação, pois, nesse tipo de rede um nó pode atuar como roteador ou transmissor de pacotes para outros nós. Esses nós estão distribuídos em um espaço e a comunicação entre eles é feita através do meio sem fio. As comunicações fim-a-fim são possíveis através de comunicações salto a salto entre o emissor e o receptor da informação. Assim, por ser um tipo de rede diferente das tradicionais, as redes *ad hoc* tem despertado grande interesse de pesquisa, e uma área bastante emergente é a modelagem de uma rede *ad hoc* como uma rede complexa. No trabalho de Krause et al. (2004), os autores apresentam as redes *ad hoc* como uma rede complexa. Nesse modelo, os N nós podem assumir posições aleatórias na rede $(x,y) \in [0,L] \times [0,L]$, onde L representa o valor limite para as variáveis x e y (este modelo é equivalente ao URP descrito na seção 1.5.1). A modelagem de ligação entre os nós sensores nesse tipo de rede está relacionada com alguns fatores como a potência do rádio do nó sensor e a existência de interferência no meio. Esses dois fatores determinam o raio de transmissão de um sensor, e dois nós estão conectados quando se encontram no raio de transmissão um do outro. Por isso, nesse mesmo trabalho são propostos alguns modelos de redes que modelam a ligação entre os nós de maneira diferente.

- **Rede com potência de transmissão constante:** Esse é um modelo no qual a potência de transmissão P é a mesma para o todos os nós sensores da rede. Considera-se também que todas as ligações são bidirecionais. Assim, se o valor de P é pequeno, a rede não é fortemente conectada. Se o valor P é muito grande a rede é fortemente conectada. Entretanto, nessa configuração o controle de acesso ao meio é prejudicado e a probabilidade de colisão em uma transmissão tende a ser grande. A escolha de P depende diretamente do tamanho da rede.

- **Rede com valor mínimo de grau para os nós:** Nesse modelo, é definido a priori um valor para o grau mínimo k_{min} de cada nó sensor. Em seguida, todos os nós enviam pacotes do tipo "Hello" e "Hello-reply" com o intuito de descobrir diretamente quem são os vizinhos, definindo assim suas ligações. Assim, o nó ajusta a sua potência de transmissão de forma a alcançar pelo menos os k_{min} vizinhos seus.

- **Rede com o grau alvo constante:** Nsse tipo de modelo, é definido um valor para o grau alvo k_{alvo} que cada nó deve possuir. Assim, os nós ajustam a potência de seus rádios de forma que cada nó tenha exatamente k_{alvo} vizinhos. Um dos requisitos é

garantir a forte conectividade entre os nós da rede. Essa garantia de conectividade exclui algumas topologias da rede, por apresentarem nós desconexos.

- **Rede com o grau alvo *scale-free*:** Esse modelo se baseia a distribuição de graus *scale-free*. Essa é uma distribuição na qual os nós são independentes e a escolha de seu grau k_i^{alvo} é feita de forma aleatória dentro de um intervalo.

Nesse contexto, alguns trabalhos são encontrados na literatura que utilizam essa abordagem para resolver problemas no cenário das redes *ad hoc*. Assim, em (Yen & Cheng 2005) é apresentado um trabalho que modela uma rede ad-hoc como uma rede complexa, com o intuito de estabelecer uma relação entre coeficiente de clusterização da rede e a quantidade de terminais escondidos na rede. A existência desses terminais implica na ocorrência de colisões nas transmissões, o que compromete o desempenho do protocolo CSMA. Assim, para o cálculo do coeficiente de clusterização foi assumido um modelo de localização uniforme dos nós e, para a cobertura foi utilizada a convenção de Torus. Após a modelagem analítica verificou-se que o número de terminais escondidos está relacionado ao número de nós na rede e a probabilidade de existir um link entre cada um deles.

Dando continuidade aos trabalhos que consideram as redes *ad hoc* como redes complexas, em Danon et al. (2005) é apresentada uma avaliação das recentes abordagens encontradas na literatura para identificação de comunidades em temos da sensibilidade na busca das comunidades e do custo computacional. Esses métodos são apresentados e classificados. Em seguida, é comparado o desempenho dos métodos aplicados a um cenário de redes *ad hoc*, no qual a estrutura de comunidades é conhecida. Assim, é apresentada uma análise da fração de comunidades corretamente identificadas para cada abordagem, como também o custo computacional de cada uma delas. Verificou-se que os algoritmos que apresentam melhor desempenho conhecem a priori a quantidade de comunidades existentes e apenas as identifica, o que sugere a necessidade de projetos de novas abordagens que primam pela redução do tempo para realizar essa identificação e que aumentem a exatidão da busca.

1.5.2.4. Redes de sensores sem fios

As redes de sensores sem fio (RSSF) são um tipo especial de redes ad-hoc composta de dispositivos sensores que atuam em conjunto no sentido de monitorar, instrumentar e, eventualmente controlar aspectos do mundo físico. As RSSFs apresentam diversas restrições no consumo de energia, dado que são compostas de dispositivos autônomos que funcionam alimentados por baterias e que a vida útil dessa bateria varia, principalmente, com a quantidade de dados transmitido pelo sensor. Dessa forma, em uma determinada rede, alguns sensores podem deixar de funcionar devido ao descarregamento da bateria enquanto outros sensores ainda estão aptos a funcionar. Alguns sensores também podem apresentar falhas e deixarem de funcionar enquanto outros sensores permanecem em pleno funcionamento. O ambiente monitorado e o fenômeno podem mudar constantemente fazendo com que a rede tenha que se adaptar a essas mudanças. Devido aos diversos desafios presentes nessas redes, há um esforço crescente de pesquisa desde o surgimento das RSSFs no final da década de 1990.

Tabela 1.2. Exemplo de tabela de roteamento para um nó D em Pásztor et al. (2010)

Líder	Alvo	Próximo	Distância
A	Sim	*Sink*	2
B	Não	C	2
C	Sim	C	1
D	Sim	–	–

Neste contexto, Pásztor et al. (2010) apresentam um esquema de reprogramação seletiva de redes de sensores móveis que monitoram alguns aspectos da vida de animais. Esta abordagem aproveita a estrutura de comunidade entre os indivíduos que estão sendo monitorado para disseminação de código de maneira eficiente. O objetivo desta aplicação é monitorar diferentes aspectos de comportamento através da implantação de nós sensores nos próprios indivíduos. Neste cenário, os programas de aplicação utilizados em cada nó sensor dependem do comportamento do animal o qual o sensor está localizado. Suponhamos que estamos interessados em estudar o comportamento de diferentes espécies de texugos[1]. Certamente vai existir um subconjunto que estará com maior frequência nas tocas, e neste caso estaremos interessados em estudar o ambiente nas tocas. Por outro lado, aqueles que não frequentam muito as tocas estarão percorrendo caminhos na floresta, e neste outro caso podemos estar interessados em conhecer os fatores (o clima, por exemplo) que determinam a escolha de um caminho ou outro.

Dado que é inviável do ponto de vista prático reprogramar os nós sensores capturando os animais, a reprogramação *online* neste caso é a única solução aceitável. Notemos que nesta situação é necessário identificar o comportamento para determinar a aplicação que será necessária num determinado nó. Para isso, os autores propõem um algoritmo de deteção de comunidades totalmente distribuído. O sistema consta de três passos:

1. o código de aplicação e as restrições que determinam a reprogramação é enviado ao nó *sink*;

2. é feito um *broadcast* das restrições para todos os nós da rede;

3. o código é disseminado, através de um algoritmos ciente da estrutura social, somente para os nós que satisfazem as restrições.

Na fase de disseminação, os indivíduos socialmente centrais são utilizados como disseminadores, já que potencialmente tendem a conhecer maior quantidade de indivíduos, satisfazendo restrições. Estes líderes formam uma hierarquia de roteamento, e as tabelas de roteamento têm a forma ilustrada na tabela 1.2. Nesta tabela, a coluna “Alvo” indica se há algum nó na comunidade, correspondente ao líder da linha da tabela, que satisfaz a restrição.

[1]animais de pernas curtas e atarracados, carnívoros que pertencem à família dos mustelídeos (Mustelidae, a mesma família de mamíferos dos furões, doninhas, lontras, e muitos outros tipos de carnívoros). Fonte: Wikipedia.

Direcionamos aos leitores interessados em mais detalhes a ler o artigo de Pásztor et al. (2010) e as referências incluídas nesse texto.

Outra aplicação interessante é o atendimento de requisições em redes de sensores sem fios. Como atender requisições quando estamos frente a uma rede dinâmica sem nenhum sistema de localização? Existem duas amplas categorias de soluções na literatura:

1. *flooding* ou *flooding* controlado: cada nó (re)transmite o pacote de requisição uma única vez;
2. caminhos aleatórios: o caminho começa em um nó fixo, e em cada paso o pacote de requisição é enviado a um vizinho. No caso mais simples esse vizinho é escolhido aleatoriamente de maneira uniforme dentre a vizinhança atual. Esse método é conhecido como "caminho aleatório simples".

As estratégias de *flooding* são ótimas em termos da latência (ou quantidade de saltos), mas têm o problema de não serem eficientes em termos de energia, aspecto fundamental para o projeto de redes de sensores sem fios. Por outro lado, as técnicas de caminhos aleatórios simples são melhores pois poupam energia permitindo que o pacote de requisição somente seja repassado para um único nó. Porém, esta técnica possui perda de desempenho em relação à latência quando o dados de interesse estão muito afastados do nó *sink*, já que esses caminhos podem conter ciclos.

Zuniga et al. (2010) apresentam uma combinação de estas duas técnicas chamada "caminho aleatório sem repetição". Nesta forma de caminho aleatório, o caminho começa com um nó fixo e em cada passo o vizinho com menor quantidade de visitas é escolhido para repassar o pacote de requisição. Em caso de mais de um nó te a mesma mínima quantidade de visitas, o próximo nó é selecionado dentre esse conjunto de maneira aleatória uniforme.

Dois aspectos podem ser estudados para avaliar o desempenho deste novo algoritmo:

- tempo de cobertura: o número esperado de passos para um caminho começado em u visitar a rede toda;
- tempo de acerto: o número esperado de passos para um caminho começado em u visitar por primeira vez v.

Os resultados apresentados por Zuniga et al. (2010) confirmam que o mecanismo proposto aumenta a probabilidade de achar nós não visitados, e consequentemente o tempo de cobertura é menor. Para mais detalhes sobre este trabalho, os leitores devem continuar a leitura de Zuniga et al. (2010) e das referências contidas nesse trabalho.

Helmy (2003) estudou o efeito da propriedade de *small world* no contexto das redes de sensores sem fio. Em seu estudo, ele classifica as redes de sensores sem fio como grafos espaciais (grafos geográficos) que tendem a apresentar maior quantidade de agrupamentos e comprimento dos caminhos médios maiores quando comparados com os grafos aleatórios. Dessa maneira, ele observa que a adição de poucas interconexões de

longa distância (chamadas de atalhos) pode reduzir drasticamente o comprimento médio dos caminhos mínimos e, assim, apresentar características de *small world*. Ele defende que esses atalhos não precisam ser totalmente aleatórios (o que seria impraticável em redes geográficas), mas podem estar confinados a um número limitado de saltos. Neste trabalho, são destacadas algumas propriedades interessantes do grafo apresentar a propriedade de *small world* como por exemplo a diminuição da latência para (os caminhos ficam menores). Os autores sugerem que alguns sensores podem aumentar suas potências de transmissão para conseguir alcançar o efeito desejado.

Nessa linha, Guidoni et al. (2008) propõem e avaliam dois modelos para geração de redes de sensores sem fio que apresentam a propriedade *small world*. Eles sugerem o uso de duas categorias de sensores, L e H, sendo o segundo possui maior capacidade de energia, processamento, memória e potência de transmissão. Dessa mameira apenas uma pequena porcentagem do total de nós da rede seria composta de sensores H (sensores mais caros). Os autores mostram que redes de sensores produzidas através desse modelo apresentam bom desempenho com relação ao consumo de energia e latência. Os modelos utilizados para geração de redes *small world* são adequações dos modelos apresentados na seção 1.4.2 para o contexto de redes de sensores. O leitor interessado deve consultar o trabalho (Guidoni et al. 2008) e as referências ali contidas para mais detalhes.

Maia et al. (2009) propuseram um protocolo para reprogramação para redes de sensores sem fio que aproveita os benefícios da propriedade *small world*. Os autores utilizam um modelo heterogêneo de redes de sensores sem fio semelhante ao do trabalho de Guidoni et al. (2008), e a tarefa de reprogramação aproveita esta infraestrutura de comunicação para reprogramar os nós sensores. Os autores demonstram que utilizando o protocolo $OAP-SW$ proposto, a tarefa de reprogramação de sensores se torna mais eficiente em termos de quantidades de mensagens necessárias, o tempo e o consumo de energia necessários para reprogramar a rede.

Nem toda métrica de centralidade presente na teoria de redes complexas (ver seção 1.3.3) é apropriada para todo contexto, em Ramos et al. (2010) é proposto o *sink betweenness* (SBet), variação do *shortest-path betweenness* que considera os menores caminhos entre nós sensores e nó *sink*. Formalmente SBet, é definido como:

$$SB(v) = \sum_{\substack{v,t \in \mathbf{V} \\ v \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}},$$

onde s é o *sink*, σ_{st} é o número de caminhos mínimos entre o *sink* e o nó t, e $\sigma_{st}(v)$ é o número de caminhos mínimos entre o *sink* e o nó t que passam pelo nó v.

Os autores desenvolvem um estudo estatístico e demonstram que para diversos cenários típicos das redes de sensores sem fio o *sink betweenness* se mostra bastante correlacionado com o consumo de energia devido à tarefa de retransmissão de pacotes (oriunda do roteamento). Dessa maneira, os autores concluem que o *sink betweenness*, uma vez estimado, pode ser utilizado no projeto de algoritmos de roteamento e de camada de enlace para, por exemplo, aliviar o efeito de sobrecarga de trabalho dos nós mais próximos ao *sink*.

Para mostrar a aplicabilidade desta métrica no contexto de redes de sensores sem

fio, o trabalho de Oliveira et al. (2010) propõe dois algoritmos de roteamento, um baseado em árvore e outro *gossip*. Com o intuito de gerar árvores de roteamento que contenham menor quantidade de arestas, e assim economizar energia, cada nó escolhe como seu pai na árvore àquele com maior centralidade. Esta abordagem baseia-se na hipótese de que os nós de maior centralidade tendem a agregar o maior número de rotas e, assim, favorecer a fusão de dados (Nakamura et al. 2007, 2009).

Por outro lado, no roteamento *gossip* proposto, a probabilidade de envio é relacionada a quão central é um nó e, com base nisto, o intuito foi de se evitar os nós mais centrais. Neste caso, o principal objetivo é melhorar o balanceamento de carga fazendo com que os nós mais centrais não sejam sobrecarregados.

Os resultados de simulação mostram ganho ao se utilizar o *sink betweenness* no roteamento quando se comparado a alguns algoritmos clássicos de roteamento. No roteamento em árvore, apresentou-se diminuição do número de arestas da árvore de roteamento, e, para o algoritmo *gossip* é mostrada uma melhor distribuição de carga de roteamento.

1.6. Conclusão

Neste trabalho foi apresentada uma introdução à teoria das redes complexas, suas principais métricas de caracterização e seus modelos mais comumente empregados em aplicações de redes tecnológicas como as redes de computadores. Também foi apresentada uma introdução aos métodos estatísticos imprescindíveis ao desenvolvimento de estudos que envolvam as técnicas discutidas nesse trabalho. Por fim foram apresentadas diversas aplicações da teoria das redes complexas no contexto de redes de computadores, mais especificamente para (i) Internet, (ii) redes P2P, (iii) redes ad-hoc e (iv) redes de sensores sem fio.

Um grande benefício que a teoria de redes complexas tem a oferecer na modelagem de redes de computadores está na caracterização da topologia através de um conjunto finito de medidas. Desse conjunto de medidas, muitas delas podem ser caras de se obter num modelo distribuído. Nesse caso, podemos lançar mão de métodos estatísticos para dar suporte aos diferentes modelos de redes complexas. Esses métodos podem ser utilizados para criar estimadores que, sob uma família de cenários com algumas propriedades comuns, descrevem propriedades mais fortes possivelmente a um menor custo computacional.

Apesar de não haver intenção de esgotar o tema nem a lista de aplicações em que já foram empregadas as técnicas de redes complexas, o presente trabalho discute o potencial que essa teoria tem apresentado para o contexto de modelagem das redes de computadores, principalmente considerando a escala dessas redes no futuro.

Referências

Adamic, L., Lukose, R., Puniyani, A. & Huberman, B. (2001), 'Search in power-law networks', *Physical Review E* **64**(46135).

Akoglu, L. & Faloutsos, C. (2009), Rtg: A recursive realistic graph generator using random typing., *in* W. L. Buntine, M. Grobelnik, D. Mladenic & J. Shaw-Taylor, eds,

‘ECML/PKDD (1)’, Vol. 5781 of *Lecture Notes in Computer Science*, Springer, pp. 13–28.

Albert, R., Jeong, H. & Barabási, A. (2000), ‘Error and attack tolerance of complex networks’, *Nature* **406**(6794), 387–482.

Albert, R., Jeong, H. & Barabási, A.-L. (1999), ‘Diameter of the World Wide Web’, *Nature* **401**, 130–131.

Anthonisse, J. M. (1971), The rush in a directed graph, Technical Report BN 9/71, Stichting Mathematisch Centrum, 2e Boerhaavestraat 49 Amsterdam.

Baddeley, A. (2007), Spatial point processes and their application, *in* W. Weil, ed., ‘Stochastic Geometry’, Vol. 1892 of *Lecture Notes in Mathematics*, Springer, Berlin, pp. 1–75.

Barabási, A. & Albert, R. (1999), ‘Emergence of scaling in random networks’, *Science* **286**, 509–512.

Berthelsen, K. K. & Møller, J. (2002), ‘A primer on perfect simulation for spatial point processes’, *Bulletin of the Brazilian Mathematical Society* **33**(3), 351–367.

Bolobás, B. (2001), *Random Graphs*, 2nd edn, Academic Press.

Brandes, U. (2001), ‘A faster algorithm for betweenness centrality’, *Journal of Mathematical Sociology* **25**, 163–177.

Brandes, U. & Pich, C. (2007), Centrality estimation in large networks, *in* ‘International Journal of Bifurcation and Chaos, Special Issue on Complex Networks? Structure and Dynamics’, Vol. 17, pp. 2303–2318.

Brin, S. (1998), The anatomy of a large-scale hypertextual web search engine, *in* ‘Computer Networks and ISDN Systems’, Vol. 30, pp. 107–117.
URL: *http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.109.4049*

Burt, R. (1995), *Structural Holes: The Social Structure of Competition*, Harvard University Press.
URL: *http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0674843711*

Bustos, O. H. & Frery, A. C. (1992), ‘Reporting Monte Carlo results in statistics: suggestions and an example’, *Revista de la Sociedad Chilena de Estadística* **9**(2), 46–95.
URL: *http://sites.google.com/site/acfrery*

Camacho, J., Guimerà, R. & Nunes Amaral, L. A. (2002), ‘Robust patterns in food web structure’, *Physical Review Letters* **88**(22), 228102.

Carchiolo, V., Malgeri, M., Mangioni, G. & Nicosia, V. (2008), ‘Emerging structures of p2p networks induced by social relationships’, *Computer Communications* **31**(3), 620–628.

Chakrabarti, D. & Faloutsos, C. (2006), 'Graph mining: Laws, generators, and algorithms', *ACM Computing Surveys* **38**(1), 2.

Chambers, J. M. (2008), *Software for Data Analisys: Programming with R*, Statistics and Computing, Springer.

Chen, Q. & Shi, D. (2004), 'The modeling of scale-free networks', *Physica A: Statistical Mechanics and its Applications* **335**(1-2), 240 – 248.

Cipra, B. A. (2000), 'The best of the 20th century: Editors name top 10 algorithms', *SIAM News* **33**(4), 1–2.

Clauset, A., Shalizi, C. R. & Newman, M. E. J. (2007), Power-law distributions in empirical data. code available at http://www.santafe.edu/ aaronc/powerlaws/.
URL: *http://arxiv.org/pdf/0706.1062*

Costa, L., Oliveira Jr., O. N., Travieso, G., Rodrigues, F. A., Villas Boas, P. R., Antiqueira, L., Viana, M. P. & da Rocha, L. E. C. (2008), Analyzing and modeling real-world phenomena with complex networks: A survey of applications.
URL: *http://arxiv.org/abs/0711.3199*

Costa, L., Rodrigues, F. A., Travieso, G. & Villas Boas, P. R. (2007), 'Characterization of complex networks: a survey of measurements', *Advances in Physics* **56**, 167–242.

Cox, D. R. & Snell, E. J. (1968), 'A general definition of residuals (with discussion)', *Journal of the Royal Statistical Society B* **30**, 248–275.

Danon, L., Diaz-Guilera, A., Duch, J. & Arenas, A. (2005), 'Comparing community structure identification', *Journal of Statistical Mechanics-Theory and Experiment* p. 10.

Díaz-Emparanza, I. (1996), Selecting the number of replications in a simulation study, Econometrics 9612006, EconWPA.
URL: *http://ideas.repec.org/p/wpa/wuwpem/9612006.html*

Dongarra, J. & Sullivan, F. (2000), 'Guest editors' introduction: The top 10 algorithms', *Computing in Science and Engineering* **2**(1), 22–23.

Du, N., Faloutsos, C., Wang, B. & Akoglu, L. (2009), Large human communication networks: patterns and a utility-driven generator, *in* 'KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, New York, NY, USA, pp. 269–278.

Erdős, P. & Rényi, A. (1959), 'On radom graph', *Publicationes Mathematicae* **6**, 290–297.

Erdős, P. & Rényi, A. (1960), 'On the evolution of random graphs', *Publications of The Mathematical Institute of The Hungarian Academy of Sciences* **5**, 17–61.

Everett, M. & Borgatti, S. P. (2005), 'Ego network betweenness', *Social Networks* **27**(1), 31 – 38.
URL: *http://www.sciencedirect.com/science/article/B6VD1-4F29STG-1/2/acad221427b2d051625939c7b00b8d97*

Everitt, B. S. (2005), *An R and S-PLUS Companion to Multivariate Analysis*, Springer-Verlag.

Faloutsos, M., Faloutsos, P. & Faloutsos, C. (1999), On power-law relationships of the internet topology, *in* 'SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication', ACM, New York, NY, USA, pp. 251–262.

Fiedler, M. (1973), 'Algebraic connectivity of graphs', *Czechoslovak Mathematical Journal* **23**(98), 298–305.

Freeman, L. C. (1977), 'A set of measures of centrality based on betweenness', *Sociometry* **40**(1), 35–41.

Frery, A. C., Ramos, H., Alencar-Neto, J. & Nakamura, E. F. (2008), Error estimation in wireless sensor networks, *in* 'ACM Symposium on Applied Computing', ACM, Fortaleza, CE, Brazil, pp. 1923–1927.

Gastner, M. T. & Newman, M. E. (2006), 'The spatial structure of networks', *The European Physical Journal B - Condensed Matter and Complex Systems* **49**(2), 247–252.

Geisberger, R., Sanders, P. & Schultes, D. (2008), Better approximation of betweenness centrality, *in* 'Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments'.
URL: *http://www.siam.org/proceedings/alenex/2008/alenex08.php*

Girvan, M. & Newman, M. E. J. (2002), 'Community structure in social and biological networks', *Proceedings of the National Academy of Sciences of the United States of America* **99**(12), 7821–7826.

Gross, J. L. & Yellen, J. (2003), *Handbook of Graph Theory - Discrete Mathematics and Its Applications*, 1 edn, CRC.

Guclu, H., Kumari, D. & Yuksel, M. (2008), Ad hoc limited scale-free models for unstructured peer-to-peer networks, *in* 'P2P '08: Proceedings of the 2008 Eighth International Conference on Peer-to-Peer Computing', IEEE Computer Society, Washington, DC, USA, pp. 160–169.

Guidoni, D. L., Mini, R. A. F. & Loureiro, A. A. F. (2008), On the design of heterogeneous sensor networks based on small world concepts, *in* 'MSWiM '08: Proceedings of the 11th ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems', ACM, pp. 309–314.

Helmy, A. (2003), 'Small worlds in wireless networks', *IEEE Communications Letters* **7**(10), 490–492.

Holme, P. & Kim, B. (2002), 'Growing scale-free networks with tunable clustering', *Physical Review E* **65**(026107).

Hui, K. Y. K., Lui, J. C. S. & Yau, D. K. Y. (2006), 'Small-world overlay p2p networks: construction, management and handling of dynamic flash crowds', *Comput. Netw.* **50**(15), 2727–2746.

James, B. (1981), *Probabilidade: um Curso em Nível Intermediário*, Projeto Euclides, Instituto de Matemática Pura e Aplicada, Rio de Janeiro.

Janson, S., Luczak, T. & Rucinski, A. (1999), *Random Graphs*, 1st edn, John Wiley.

Keller, E. F. (2005), 'Revisiting ldquoscale-freerdquo networks', *BioEssays* **27**(10), 1060–1068.

Kleinberg, J. (2000), The small-world phenomenon: an algorithm perspective, *in* 'STOC '00: Proceedings of the thirty-second annual ACM symposium on Theory of computing', ACM, New York, NY, USA, pp. 163–170.

Kleinberg, J. M. (1999), 'Authoritative sources in a hyperlinked environment', *Journal of the ACM* **46**(5), 604–632.
URL: *http://dx.doi.org/10.1145/324133.324140*

Krause, W., Glauche, I., Sollacher, R. & Greiner, M. (2004), 'Impact of network structure on the capacity of wireless multihop ad hoc communication', *Physica A - Statistical Mechanics and Its Applications* **338**(3-4), 633–658.

Krause, W., Scholz, J. & Greiner, M. (2006), 'Optimized network structure and routing metric in wireless multihop ad hoc communication', *Physica A: Statistical Mechanics and Its Applications* **361**(2), 707–723.

Leskovec, J., Backstrom, L., Kumar, R. & Tomkins, A. (2008), Microscopic evolution of social networks, *in* 'KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, New York, NY, USA, pp. 462–470.

Leskovec, J., Kleinberg, J. & Faloutsos, C. (2007), 'Graph evolution: Densification and shrinking diameters', *ACM Transaction on Knowledge and Discovery from Data* **1**(1), 2.

Li, L., Alderson, D., Doyle, J. & Willinger, W. (2005), 'Towards a theory of scale-free graphs: Definition, properties, and implications', *Internet Mathematics* **2**(4), 431–523.

Lilijeros, F., Edling, C., Amaral, L., Stanley, E. & åberg, Y. (2001), 'The web of human sexual contacts', *Nature* **411**, 907–908.

Maia, G., Guidoni, D. L., Aquino, A. L. & Loureiro, A. A. (2009), Improving an over-the-air programming protocol for wireless sensor networks based on small world concepts, *in* 'MSWiM '09: Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems', ACM, New York, NY, USA, pp. 261–267.

McGlohon, M., Akoglu, L. & Faloutsos, C. (2008), Weighted graphs and disconnected components: patterns and a generator, *in* 'KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, New York, NY, USA, pp. 524–532.

Milgran, S. (1967), 'The small world problem', *Psycology Today* pp. 60–67.

Mitzenmacher, M. (2004), 'A brief history of generative models for power law and lognormal distributions', *Internet Mathematics* **1**(2), 226–251.

Møller, J. & Waagepetersen, R. P. (2007), 'Modern statistics for spatial point processes', *Scandinavian Journal of Statistics* **34**(4), 643–684.

Motter, A., Nishikawa, T. & Lai, Y. (2002), 'Range-based attacks on links in scale-free networks: Are long-range links responsible for the small-world phenomenon?', *Physical Review E* **66**(065103).

Nakamura, E. F., Loureiro, A. A. F. & Frery, A. C. (2007), 'Information fusion for wireless sensor networks: Methods, models, and classifications', *ACM Computing Surveys* **39**(3), 9.

Nakamura, E. F., Ramos, H. S., Villas, L. A., de Oliveira, H. A. B. F., de Aquino, A. L. L. & Loureiro, A. A. F. (2009), 'A reactive role assignment for data routing in event-based wireless sensor networks', *Computer Networks* **53**(12), 1980–1996.

Newman, M. (2003), 'The structure and function of complex networks', *SIAM Review* **45**(2), 167–256.

Newman, M. E. (2001), 'The structure of scientific collaboration networks.', *Proc Natl Acad Sci U S A* **98**(2), 404–409.

Newman, M. E. J. & Watts, D. J. (1999), 'Scaling and percolation in the small-world network model', *Physical Review E* **60**(6), 7332–7342.

Oliveira, E. M. R., Ramos, H. S. & A.F. Loureiro, A. (2010), Centrality-based routing for wireless sensor networks, *in* 'ICCCN 2010 Track on Wireless Networks and Emerging Technologies (WNET)', Zurich, Switzerland. Artigo submetido, em processo de avaliação.

Pásztor, B., Motolla, L., Mascolo, C., Picco, G. P., Ellwood, S. & Macdonald, D. (2010), Selective reprogramming of mobile sensor networks through social community detection, *in* 'Proceedings of the 7th European Conference of Wireless Sensor Networks (EWSN)', Vol. 5970 of *Lecture Notes in Computer Science*, pp. 178–193.

R Development Core Team (2009), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. **URL:** *http://www.R-project.org*

Ramos, H. S., Frery, A. C., Almiron, M. G., Oliveira, E. M. & Loureiro, A. A. F. (2010), 'A new topological measure of centrality and its application to the energy hole problem in wireless sensors networks', *Special Issue of Computer Communications on Complex Networks* . Artigo submetido, em processo de avaliação.

Robert, C. P. & Casella, G. (2000), *Monte Carlo Statistical Methods*, Springer Texts in Statistics, Springer, New York.

Sarshar, N. & Roychowdhury, V. (2004), 'Scale-free and stable structures in complex ad hoc networks', *Physical Review E* **69**(026101).

Shimbel, A. (1953), 'Structural parameters of communication networks', *Bulletin of Mathematical Biology* **15**(4), 501–507.
URL: *http://dx.doi.org/10.1007/BF02476438*

Siganos, G., Faloutsos, M., Faloutsos, P. & Faloutsos, C. (2003), 'Power laws and the AS-level internet topology', *IEEE-ACM Transactions on Networking* **11**(4), 514–524.

Solomonoff, R. & Rapoport, A. (1951), 'Connectivity of random nets', *Bulletin of Mathmatical Biophysics* **13**, 107–117.

Tselishchev, Y., Boulis, A. & Libman, L. (2010), Experiences and lessons from implementing a wireless sensor network mac protocol in the castalia simulator, *in* 'IEEE Wireless Communications & Networking Conference 2010 (WCNC 2010)', Sydney/Australia.

Valente, T. W. & Foreman, R. K. (1998), 'Integration and radiality: Measuring the extent of an individual's connectedness and reachability in a network', *Social Networks* **20**(1), 89–105.
URL: *http://www.sciencedirect.com/science/article/B6VD1-3SX6NDT-5/2/6195078c6b2163bb7bc8a98caf174973*

Vázquez, A. (2003), 'Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations', *Physical Review E* **67**(5), 056104+.

Wang, X. F. & Chen, G. (2003), 'Complex networks: small-world, scale-free and beyond', *Circuits and Systems Magazine, IEEE* **3**(1), 6–20.

Watts, D. J. & Strogatz, S. H. (1998), 'Collective dynamics of 'small-world' networks', *Nature* **393**(6684), 440–442.

Yen, L. H. & Cheng, Y. M. (2005), 'Clustering coefficient of wireless ad hoc networks and the quantity of hidden terminals', *IEEE Communications Letters* **9**(3), 234–236.

Zhou, H. & Lipowsky, R. (2005), 'Dynamic pattern evolution on scale-free networks', *Proceedings of the National Academy of Sciences USA* **102**(29), 10052–10057.

Zhou, S. & RJ, R. M. (2004), 'The rich-club phenomenon in the Internet topology', *IEEE Communications Letters* **8**(3), 180 – 182.

Zuniga, M., Avin, C. & Hauswirth, M. (2010), Querying dynamic wireless sensor networks with non-revisiting random walks, *in* 'Proceedings of the 7th European Conference of Wireless Sensor Networks (EWSN)', Vol. 5970 of *Lecture Notes in Computer Science*, pp. 49–64.

Capítulo 2

Interconexão de Redes na Internet do Futuro: Desafios e Soluções

Miguel Elias M. Campista, Lyno Henrique G. Ferraz, Igor M. Moraes,
Marcelo Luiz D. Lanza, Luís Henrique M. K. Costa e
Otto Carlos M. B. Duarte

GTA/PEE/COPPE/DEL-Poli - UFRJ - Rio de Janeiro, Brasil

Abstract

This short-course presents the main challenges for network interconnection in the Future Internet. We introduce the shortcomings of the Internet current model and the main proposals to solve them. Among all shortcomings, many are consequence of unforeseen demands by the Internet original design such as: mobility, multi-homing, multi-path, and network scalability. These challenges have been attracting many research efforts in the last few years because of their relevance and complexity. In this short-course, new protocols for network interconnection are presented, from IP extensions to radical proposals to replace IP in the Future Internet. In addition, we introduce new intra- and interdomain routing protocols as well as experimental results obtained with one of these proposals prototypes.

Resumo

Este minicurso apresenta os principais desafios para a camada de interconexão de redes da Internet do Futuro. As limitações do modelo atual da Internet e as principais propostas para solucioná-las são apresentadas. Dentre os desafios, muitos são consequências de demandas não previstas pelo projeto inicial da Internet, como: mobilidade, redes multidomiciliadas, múltiplos caminhos e escalabilidade da rede. Esses desafios têm atraído muito investimento em pesquisa nos últimos anos devido à relevância e à complexidade do tema. Neste minicurso, são apresentados novos protocolos para interconexão de redes, desde extensões do IP até propostas radicais para substituir o IP na Internet do Futuro. Além disso, propostas para novos protocolos de roteamento intra e interdomínio são apresentadas, bem como resultados práticos obtidos com um dos protótipos dessas propostas.

2.1. Introdução

No início dos anos 80, a Internet surgia com a adoção da pilha de protocolos TCP/IP. Naquela época, a Internet era composta por estações fixas de grande porte cujas principais aplicações eram o acesso remoto e a troca de arquivos. Nesse cenário, o IP (*Internet Protocol*) tinha como objetivo atender requisitos fundamentais como interconectar redes, prover conectividade fim-a-fim e garantir acessibilidade global [Clark et al. 2004]. Esses requisitos deviam ser atendidos seja qual fosse a aplicação e a tecnologia de controle de acesso ao meio utilizada. Além do IP, os protocolos de roteamento também eram utilizados na interconexão de redes. Esses protocolos cumpriam o mesmo papel dos protocolos de roteamento atuais. Entretanto, eles podiam calcular os menores caminhos entre qualquer par origem-destino, já que a Internet era composta por poucos nós e era apenas subdividida em poucas redes de instituições sem fins lucrativos. Cada roteador, por conseguinte, conhecia o melhor caminho disponível para qualquer uma das redes da Internet. Um exemplo disso, foi o projeto do GGP (*Gateway-to-Gateway Protocol*) [Hinden e Sheltzer 1982] que foi um dos primeiros protocolos de interconexão de redes. O GGP listava em suas mensagens de atualização as distâncias em número de saltos para todas as redes da Internet e suportava até no máximo 256 redes.

Nos anos seguintes, o número de usuários e de redes na Internet aumentou aceleradamente. Esse aumento tornou as atualizações de topologia mais frequentes e as atualizações do GGP mais complexas, visto que cada uma das redes era administrada por grupos ou instituições diferentes. Além disso, havia mais de uma versão do GGP em uso, o que dificultava a interoperação das redes. Essas dificuldades levaram a Internet a ser dividida em Sistemas Autônomos (*Autonomous Systems* - ASes), definidos como um conjunto de redes e roteadores administrados por um grupo ou uma instituição comum. Cada Sistema Autônomo escolhe o seu próprio protocolo de roteamento interno, chamado protocolo intradomínio, e para se comunicar com redes em outros Sistemas Autônomos, utiliza um protocolo interdomínio. O protocolo interdomínio era comum a todos os ASes já que as informações de roteamento precisam ser trocadas para manutenção global da conectividade. O primeiro protocolo interdomínio desenvolvido foi o EGP (*Exterior Gateway Protocol*) [Mills 1984] que também sucumbiu devido a problemas de escalabilidade. O EGP foi substituído pelo BGP (*Border Gateway Protocol*) [Lougheed e Rekhter 1989] que se tornou o protocolo interdomínio da Internet.

A organização da Internet de hoje é a mesma que surgiu após a sua divisão em Sistemas Autônomos. Além disso, a Internet ainda deve atender todos os seus requisitos fundamentais, pois foi por isso mesmo que se tornou o sucesso de hoje. Outra característica mantida foi o crescimento do número de ASes e, consequentemente, de redes. Dados mostram que a Internet atual consiste de 7,4 milhões de redes compostas por no máximo 254 estações (redes /24). Essas redes representam 95% de todas as redes alcançáveis da Internet [Caida 2010]. Outros dados mostram que o crescimento da Internet foi de 380,3% de 2000 até 2009 [Internet World Stats 2010]. Tal crescimento é impulsionado pela oportunidade de negócios e pela popularização da Internet. A oportunidade de negócios surgiu da necessidade de trânsito de dados entre ASes não conectados diretamente. Essa necessidade levou a acordos entre ASes vizinhos que estão no caminho entre pares origem-destino para transporte de dados. Já a popularização da Internet é consequência da facilidade e rapidez na obtenção de informações proporcionadas para

usuários corporativos e residenciais que pagam os provedores de acesso pelo serviço.

Assim como ocorreu no passado, mais uma vez a Internet está chegando a um ponto onde mudanças se tornam essenciais. O sucesso da Internet não vem sem consequências. Mesmo os protocolos atuais como o BGP e o próprio IP vêm apresentando sinais de fadiga. Um indicativo disso é o BGP ter sido proposto em 1989 e ter sido atualizado até a sua versão quatro em apenas seis anos [Lougheed e Rekhter 1989, Rekhter e Li 1995]. Em 2006, o BGP-4 foi atualizado novamente para corrigir erros de edição de RFCs (*Request For Comments*) anteriores [Rekhter et al. 2006]. Embora o BGP seja fundamental, os problemas dele também são oriundos das próprias limitações do IP, que não acompanhou no mesmo passo a evolução da Internet. Portanto, além dos problemas relacionados ao crescimento da Internet, nem o projeto inicial do IP nem tampouco dos protocolos de roteamento existentes foram desenvolvidos tendo em vista a possibilidade de evoluções para atender demandas emergentes. Dentre essas demandas, as mais relevantes são o suporte à mobilidade, o suporte aos múltiplos domicílios (*multi-homing*), o suporte aos múltiplos caminhos e a escalabilidade dos roteadores. Embora novas propostas ou extensões tenham surgido como resposta às novas demandas, p. ex. o IPv6 [Deering e Hinden 1998], o IP móvel [Perkins 2002] e o multicast [Deering 1988, Costa et al. 2006], nenhuma delas solucionou o problema por completo, e em alguns casos, nem foram aplicadas na prática.

Atualmente, investiga-se qual a melhor abordagem a ser tomada naquela que será a Internet do Futuro. Dentre as possíveis, estão a ruptura total com o modelo atual da Internet e o consequente surgimento de novos protocolos para cumprir um papel semelhante ao do IP [Crowcroft 2008] ou a continuidade da elaboração de novas extensões aos protocolos [Ratnasamy et al. 2005]. A primeira abordagem pode ir de encontro aos interesses dos administradores dos ASes que já possuem base instalada operacional. Já a segunda abordagem pode apenas adiar o problema enfrentado atualmente. Um ponto importante é a possibilidade de novos protocolos incluírem todas as novas demandas emergentes ou deixarem ainda algumas como casos a parte. Em comum, pode-se destacar que a nova proposta deverá ser escalável e capaz de evoluir para atender os futuros avanços da Internet. Este capítulo apresenta os principais desafios em interconexão de redes na Internet do Futuro, bem como as limitações do modelo atual e as propostas para solucioná-las. Adicionalmente, alguns dos novos protocolos de roteamento assim como resultados experimentais são apresentados. Este capítulo está organizado da seguinte maneira. A Seção 2.2 apresenta a arquitetura atual da Internet e as definições empregadas neste capítulo. A Seção 2.3 discute os principais desafios enfrentados na Internet atual para prover serviços como: mobilidade, múltiplos domicílios, múltiplos caminhos, caminhos programáveis e escalabilidade dos roteadores. Essa seção ainda apresenta as limitações da arquitetura atual que impedem que tais serviços sejam oferecidos plenamente. A Seção 2.4 descreve os novos conceitos da área para aprimorar a interconexão de redes na Internet do Futuro, bem como suas principais propostas. A Seção 2.5 apresenta resultados experimentais obtidos com protótipos de uma das novas propostas para a Internet do Futuro. Por fim, a Seção 2.6 conclui este capítulo.

2.2. Arquitetura Atual da Internet

A arquitetura atual da Internet é composta por diferentes redes interconectadas através do IP. As redes administradas pela mesma instituição são denominadas Sistemas Autônomos (ASes). Cada AS executa um protocolo de roteamento intradomínio para interconectar seus próprios roteadores enquanto os diferentes ASes executam um protocolo de roteamento interdomínio para se comunicarem. Essa arquitetura é uma evolução da arquitetura original da Internet que era apenas subdividida em redes de instituições diferentes. A Figura 2.1 mostra a topologia da Internet no início dos anos 80 [History Museum 2010].

A posição de um AS na topologia da Internet pode ser usada para classificação. Um AS diretamente conectado aos usuários é denominado de AS de borda ou provedor de acesso. Tais ASes tarifam os usuários pelo acesso à Internet. Os usuários normalmente estão localizados em redes de acesso, que também podem ser denominadas de redes *stub*. As redes *stub* conhecem apenas um caminho para outras redes. Esses caminhos são chamados de rotas *default* já que são sempre utilizadas para alcançar qualquer outra rede. As rotas *default* são anunciadas para as estações da rede de acesso pelo ponto de interconexão (*gateway*) da rede. Uma rede de acesso recebe uma faixa de endereços IP pertencente à faixa do AS de borda. Essa faixa é utilizada para atribuição de endereços às estações da rede. Logo, o prefixo comum dos endereços IP das estações pode ser usado para identificação da rede. Da mesma maneira, um endereço IP identifica uma estação.

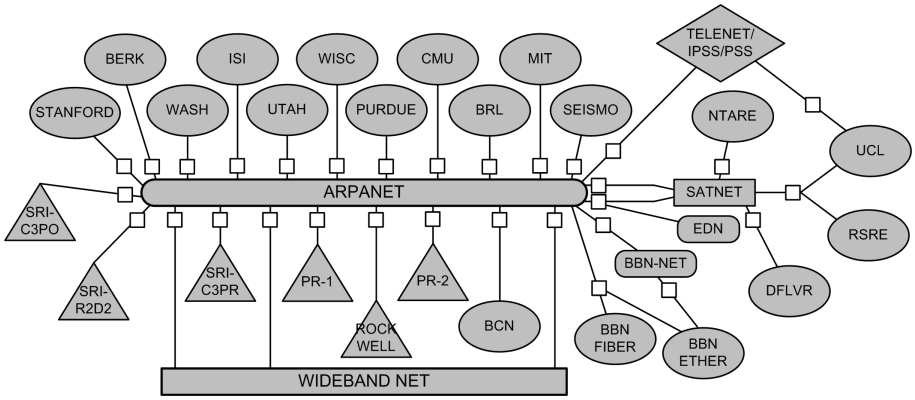


Figura 2.1. A Internet no início do anos 80.

Os ASes que não provêem acesso diretamente a usuários formam o núcleo da Internet. Esses ASes são conhecidos como ASes de trânsito por oferecerem serviços de transferência de dados entre ASes. Os ASes de trânsito são responsáveis por encaminhar todo o tráfego entre as bordas da Internet. Para isso, assume-se que toda origem e destino de tráfego estão localizados na borda da rede. Essa premissa está de acordo com um dos requisitos da Internet que é a manutenção da inteligência nas bordas para que o núcleo seja simples. Os ASes de trânsito tarifam os seus ASes vizinhos, sejam eles de borda ou outros de trânsito, dependendo do tipo de acordo estabelecido e da quantidade de tráfego injetado pelo vizinho no AS de trânsito.

As instituições responsáveis pelos ASes de trânsito são denominadas de provedores de serviço da Internet (*Internet Service Providers* - ISPs). Os ASes de trânsito normal-

mente não possuem rotas *default*. Para isso, assume-se que os ISPs possuem visão global de todos os ASes da Internet. A ausência de rotas *default* rendeu ao conjunto de ASes de trânsito o nome de zona livre de rota *default* (*Default Free Zone* - DFZ). A Figura 2.2 mostra a arquitetura atual da Internet e as suas subdivisões em Sistemas Autônomos (ASes). Os ASes de borda possuem ligadas a eles as redes de acesso dos usuários.

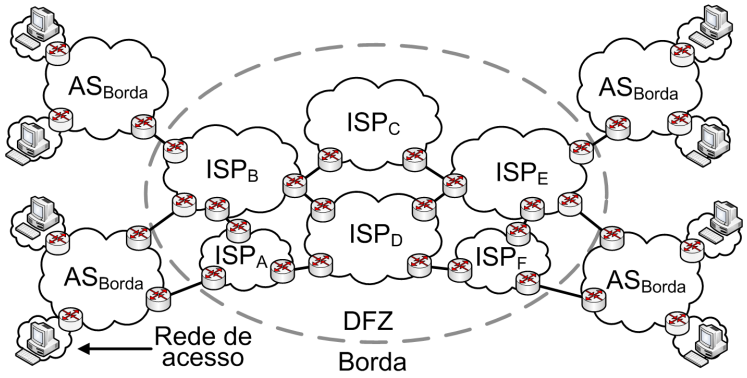


Figura 2.2. A arquitetura da Internet atual.

Duas definições importantes que serão usadas posteriormente são a FIB (*Forwarding Information Base*) e a RIB (*Routing Information Base*). A FIB e a RIB são tabelas com funções distintas utilizadas pelos roteadores da Internet. A FIB é a tabela de roteamento que é consultada sempre que o roteador recebe um pacote. Após uma consulta à tabela de roteamento, o roteador encaminha o pacote recebido pela interface de saída correspondente ao caminho escolhido para alcançar a rede de destino. A RIB, em contrapartida, é a base de dados que é utilizada para construir a FIB. Por exemplo, no protocolo OSPF (*Open Shortest Path First*), a FIB contém o mapa conhecido da topologia da rede. As FIBs são construídas para acelerar a consulta a interfaces de saída enquanto as RIBs são construídas para tornar a atualização da base de dados mais rápida. A Figura 2.3 ilustra um esquema básico de um roteador. A figura mostra o caminho percorrido pelos pacotes de controle (caminho de controle) e pelos pacotes de dados (caminho de dados).

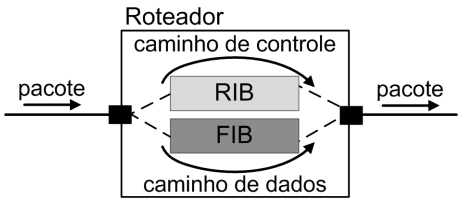


Figura 2.3. Esquema básico de um roteador atual.

2.3. Desafios em Interconexão de Redes

Muitos dos desafios enfrentados pela Internet atual estão diretamente ligados à sua arquitetura e ao crescimento acelerado do número de estações e redes. A arquitetura original da Internet não previu o suporte a serviços como mobilidade, múltiplos domicílios, múltiplos caminhos e programabilidade de caminhos. Esta seção discute os principais desafios da interconexão de redes e do roteamento na Internet do futuro.

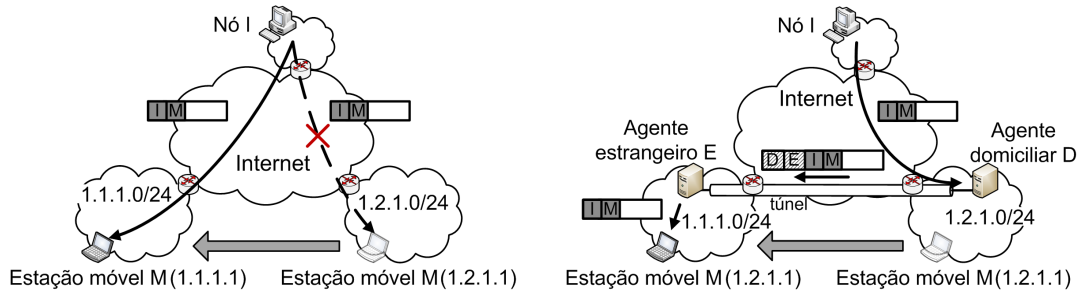
Mobilidade

A disseminação das redes sem-fio é um dos principais desafios à arquitetura original da Internet. As redes sem-fio possibilitam que as estações se movimentem. Logo, uma vez que a arquitetura original da Internet não previa a mobilidade, muitas das soluções utilizadas não se adequam a essa característica. Um exemplo típico é o funcionamento do TCP (*Transmission Control Protocol*) em redes sem-fio [Hanbali et al. 2005]. O TCP foi desenvolvido para uso na Internet cabeada. Por isso, esse protocolo considera que atrasos na recepção de reconhecimentos positivos ou perda de pacotes são devido a congestionamentos na rede já que as perdas por erro de transmissão são negligenciáveis. Assim, para evitar o colapso da rede, as fontes TCP reduzem as suas taxas de transmissão quando detectam perdas. Nas redes sem-fio essa mesma premissa pode não ser mais verdadeira visto que as perdas por erro de transmissão são consideráveis. Portanto, reduzir a taxa de transmissão representa um desperdício de banda passante uma vez que a rede não enfrenta problemas de congestionamento.

A mobilidade das estações introduz desafios para camada de interconexão de redes, em especial no que concerne o endereçamento e o roteamento. Na Internet original, o endereçamento foi organizado de maneira hierárquica para possibilitar a agregação de rotas e assim aumentar a escalabilidade do roteamento. Um exemplo dessa organização é visto nos prefixos usados pelas redes de acesso ligadas a um AS. Os prefixos das redes são faixas pertencentes ao AS de borda. Assim, o AS de borda pode anunciar aos outros ASes vizinhos os prefixos agregados das redes ligadas a ele. A consequência da agregação é a redução do número de entradas nas tabelas de roteamento e a simplificação da busca nas FIBs para encaminhar pacotes às estações.

A estrutura hierárquica da Internet levou à organização geográfica de endereços IP. Tal organização associou ao endereço IP a localização da estação na Internet. Por outro lado, a identificação de estações da Internet é feita através de nomes que são mapeados pelo DNS (*Domain Name System*) em endereços IP. Assim, o endereço IP é utilizado para localização e identificação de uma estação, o que caracteriza a "semântica sobrecarregada" do endereço IP. Esse problema representa um entrave à mobilidade das estações já que ao se moverem, as estações mudam a sua localização e, consequentemente, devem mudar o seu endereço IP. Porém, a arquitetura da Internet atual assume que os endereços IP são invariantes durante toda a comunicação entre estações. Como padrão, se uma estação muda de rede de acesso à Internet, essa estação deve reconfigurar o seu endereço IP para um que seja topologicamente correto de acordo com a sua nova rede. Ao receber um novo endereço IP, as comunicações estabelecidas com o endereço IP anterior são perdidas. Além disso, o estabelecimento de conexão com uma estação que muda constantemente de endereço dificulta a resolução de nomes já que a estação não possui um endereço permanente. A Figura 2.4(a) ilustra a reconfiguração do endereço IP da estação móvel M ao mudar de rede de acesso. A mudança do endereço provoca o restabelecimento da conexão com o nó I da Internet.

Soluções adicionais ao IP devem ser propostas para que as comunicações, e possivelmente as conexões pré-estabelecidas, sejam mantidas mesmo após o deslocamento das estações. Uma proposta inicial para resolver o problema foi o IP móvel [Perkins 2002]. O IP móvel mantém as conexões de uma estação móvel através da manutenção da associação



(a) Mudança de endereço em decorrência da mudança de rede de acesso.

(b) Funcionamento do IP móvel.

Figura 2.4. Mobilidade de estação.

dessa estação com o seu endereço IP original, mesmo após o deslocamento. Para isso, é necessário que a estação móvel informe à rede original qual a sua nova localização para que todo pacote recebido possa ser encaminhado para a estação em seu novo endereço na rede estrangeira. A nova localização da estação móvel é recebida pelo agente domiciliar (*home agent*), que é uma estação na rede de acesso original responsável por receber e encaminhar todo o tráfego destinado à estação móvel. O agente domiciliar encapsula os pacotes destinados à estação móvel e os encaminha através de um túnel estabelecido entre o agente domiciliar e uma estação da rede estrangeira. Essa estação, denominada agente estrangeiro (*foreign agent*), é responsável por atribuir um endereço IP válido da rede estrangeira à estação móvel e também é responsável por encaminhar os pacotes recebidos através do túnel até essa mesma estação. A estação móvel recebe um endereço IP válido do agente estrangeiro após um procedimento de registro. O procedimento de encaminhamento, ilustrado pela Figura 2.4(b), pode ser resumido da seguinte maneira. Todos os pacotes enviados à estação móvel M são recebidos pelo agente domiciliar D. Esse agente encapsula os pacotes e os encaminha pelo túnel até o agente estrangeiro E. O agente estrangeiro E desencapsula os pacotes e os encaminha até a estação móvel M. O tráfego no sentido reverso é originado pela estação móvel M e enviado até o agente estrangeiro E. Esse encapsula os pacotes e os envia até o agente domiciliar D através do túnel. O agente domiciliar D recebe os pacotes, os desencapsula e os envia para a Internet. Apesar de o procedimento permitir o deslocamento das estações móveis, o IP móvel encaminha os pacotes de maneira indireta entre as estações de origem e destino. Esse encaminhamento indireto é refletido em um caminho mais longo percorrido pelos pacotes, o que reduz a eficiência do roteamento na Internet.

O IPv6 [Johnson et al. 2004] móvel dispensa o uso do agente estrangeiro. Portanto, o túnel é estabelecido entre o agente domiciliar e a própria estação móvel. Além disso, o IPv6 móvel permite que os pacotes sejam encaminhados da Internet até a estação móvel sem precisar passar pela rede domiciliar. O encaminhamento direto e possível com o uso de uma extensão de cabeçalho para roteamento (*routing header extension*) do IPv6. A extensão permite que a estação móvel utilize o endereço IP da rede estrangeira como endereço de origem e liste na extensão do cabeçalho o endereço da rede domiciliar. O nó da Internet ao receber o pacote reconhece o novo endereço da estação e, após um procedimento de associação, utiliza o endereço da estação móvel como endereço de destino dos próximos pacotes. Mesmo com essas diferenças para o IP móvel, o IPv6 móvel ainda

exibe problemas como latência no restabelecimento do acesso à Internet, principalmente se a estação móvel se deslocar a altas velocidades.

Múltiplos domicílios

Os múltiplos domicílios (*multi-homing*) são redes de acesso ou estações que se conectam à Internet através de múltiplas conexões. Essas conexões são obtidas de diferentes ISPs ou do mesmo ISP, mas usando faixas de endereços diferentes. A Figura 2.5 ilustra uma estação multidomiciliada (*host multihoming*) e uma rede multidomiciliada (*site multihoming*). No primeiro caso, a estação possui interfaces configuradas com endereços IP diferentes. Já no segundo caso, o ponto de interconexão possui saídas para dois ISPs diferentes, porém as estações internas podem possuir endereços IP privados ou podem ter a sua única interface configurada com endereços IP de cada um dos ISPs. Assim, as redes multidomiciliadas podem ser identificadas por mais de uma faixa de endereços, que devem ser anunciadas para todos os ASes da Internet através de rotas do BGP. A motivação para o emprego dos múltiplos domicílios é o aumento da confiabilidade na comunicação com a Internet [Farinacci et al. 2009b, Nordmark e Bagnulo 2009]. Em caso de falha de um dos provedores, ainda há alternativas para a manutenção da comunicação. Para maior confiabilidade, as redes multidomiciliadas devem estar configuradas de maneira a isolar eventuais falhas individuais. Na ausência de falhas, as múltiplas faixas de endereços podem ser também exploradas para possibilitar engenharia de tráfego, maximizar vazão agregada e até mesmo reduzir custos. Um exemplo de redução de custos é o uso de um ISP mais caro para tráfego que exige qualidade de serviço e o uso de um mais barato, no caso contrário.

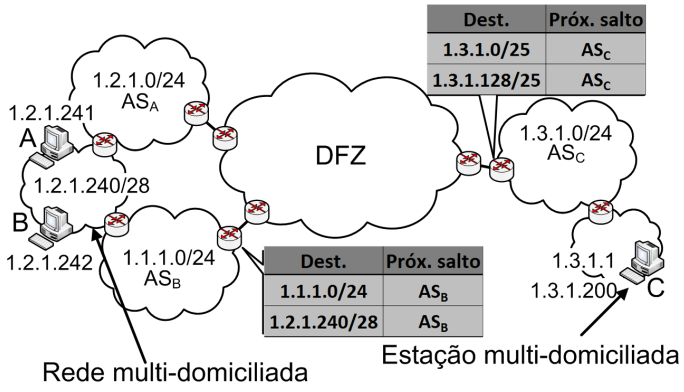


Figura 2.5. Exemplos de estação e rede multidomiciliada.

A utilização dos múltiplos domicílios traz dificuldades do ponto de vista do roteamento. As múltiplas faixas podem ser alocadas por ISPs (*provider assigned*) ou podem ser independentes do ISP (*provider independent*). Nessa última opção, a faixa de endereços é alocada diretamente pelo órgão regional de distribuição de endereços da Internet. A principal vantagem em se utilizar faixas independentes é evitar reconfiguração caso haja a troca de provedor de acesso. O funcionamento dos múltiplos domicílios viola a organização hierárquica de endereços da Internet, o que impede a agregação de endereços. Essa técnica requer o anúncio das múltiplas faixas desagregadas para que todas as outras redes na Internet tomem conhecimento da localização daquela faixa. Uma vez que cada

provedor de serviço possui suas faixas de endereços pré-estabelecidas, anunciar faixas de endereços desagregadas ou de outros provedores causa problemas de escalabilidade, como o aumento das tabelas BGP.

Múltiplos caminhos

Muitos protocolos de roteamento atuais escolhem apenas um único caminho para interconectar um par origem-destino na rede. Protocolos intradomínio como o OSPF e o RIP (*Routing Information Protocol*) tipicamente mantêm em suas tabelas de roteamento apenas o vizinho que pertence ao melhor caminho até o destino. O OSPF em especial oferece variantes que calculam múltiplos caminhos, mas que são pouco exploradas na prática. Essa limitação dificulta a introdução de confiabilidade e flexibilidade no roteamento, uma vez que seria possível dividir o tráfego entre caminhos diferentes. Outras vantagens são a possibilidade de escolher caminhos de acordo com os requisitos das aplicações, por exemplo, caminhos com maior banda ou menor atraso, e ainda evitar caminhos congestionados [He e Rexford 2008].

Os múltiplos caminhos representam uma opção pouco explorada na Internet apesar do seu potencial. Medidas demonstram que existe um caminho alternativo com taxa de perda e atraso menores que o caminho usado em 30 a 80% do tempo [He e Rexford 2008]. O emprego desses caminhos pode aumentar significativamente o desempenho do encaminhamento de pacotes na Internet. Entretanto, os múltiplos caminhos deixam de ser explorados porque muitas vezes esbarram em limitações de camadas superiores. Um exemplo disso é a possibilidade do desordenamento de pacotes de uma mesma conexão TCP se a diversidade de caminhos for explorada no nível de pacotes e não no nível de fluxos. A Figura 2.6 ilustra o exemplo do uso dos múltiplos caminhos entre as estações A e C. Os múltiplos caminhos são usados para envio de tráfego ao mesmo tempo, o que difere essa estratégia do uso dos múltiplos domicílios. A Figura 2.6 ilustra um exemplo de múltiplos caminhos totalmente disjuntos. Entretanto, os múltiplos caminhos podem ser parcialmente disjuntos, no caso de alguma rede de acesso ou AS que não ofereça múltiplas saídas para a Internet, por exemplo.

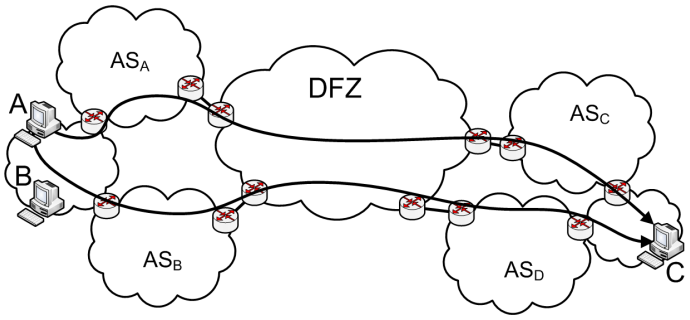


Figura 2.6. Exemplos do uso dos múltiplos caminhos na Internet.

Embora a topologia da Internet ofereça múltiplos caminhos devido às múltiplas conexões entre ASes e aos múltiplos caminhos intradomínio, problemas relacionados com escalabilidade, acordos comerciais entre os diferentes ISPs e o próprio projeto do BGP dificultam o emprego dessa possibilidade [He e Rexford 2008]. A primeira questão é uma

consequência da obrigação dos roteadores em armazenarem informações de múltiplos caminhos para cada destino. Essa característica exige a execução de algoritmos ou variações de algoritmos com complexidade superior ao Dijkstra ou ao Bellman-Ford, usados em muitos dos protocolos intradomínio, que aumenta a troca de informações de controle e ainda aumenta o número de entradas nas tabelas de roteamento [He e Rexford 2008]. A segunda questão é consequência da divisão da Internet em múltiplos ASes. Para serem plenamente explorados, os múltiplos caminhos devem ter condições de passar por redes que pertençam a diferentes organizações. Tais organizações não necessariamente possuem acordos entre si e, portanto, não encaminham tráfego vindo desses ASes. Já a última questão está relacionada com a maneira como o BGP foi desenvolvido.

O BGP, assim como os protocolos intradomínio, foi desenvolvido para ser um protocolo que ofereça apenas uma opção de caminho para cada prefixo de rede da Internet. Cada ISP anuncia aos seus vizinhos um único caminho ativo para alcançar outras redes através dele. O caminho disponibilizado depende das políticas utilizadas pelo BGP. Entretanto, se mais de um caminho fosse oferecido pelos ISPs aos seus vizinhos, parte do controle do tráfego encaminhado seria perdido. Isso ocorre pois não há como prever a distribuição de tráfego realizada pelos ISPs vizinhos que é encaminhado através das múltiplas opções oferecidas. Esse controle é importante para que os ISPs possam realizar engenharia de tráfego dentro da sua própria rede. Algumas opções para forçar os múltiplos caminhos sem a necessidade de cada ISP anunciar múltiplos caminhos é através do uso do roteamento pela fonte ou das redes sobrecamada (*overlay*). Em ambas as opções, o ISP também perde parte do controle de sua rede. Isso porque tanto a fonte que escolhe o caminho quanto as redes formadas em camadas superiores podem desconsiderar as políticas de cada ISP e as ligações na camada física, respectivamente.

Caminhos programáveis

Atualmente, os caminhos seguidos pelos pacotes na Internet são definidos por protocolos de roteamento que, em especial o BGP, envolvem muitas configurações manuais de administradores de rede. Os protocolos de roteamento intradomínio escolhem o melhor caminho entre a origem e o destino baseado em métricas estabelecidas, p. ex., o número de roteadores atravessados, e os protocolos de roteamento interdomínio consideram acordos comerciais para escolher qual dos ISPs vizinhos deve ser utilizado. Nesse último caso, as condições nas quais o tráfego é encaminhado dependem do tipo de acordo entre as partes que pode inclusive limitar requisitos de rede como a banda passante máxima.

Uma possibilidade para conceder maiores poderes aos usuários é criar critérios para que os caminhos escolhidos na Internet sejam programáveis ou configuráveis de acordo com requisitos desejados pelos próprios usuários ou por aplicações no nível de usuário. Tais requisitos podem ser demandas de uma aplicação, como por exemplo, garantias de qualidade de serviço, que poderiam ser escolhidas dinamicamente por agentes inteligentes ou pelos usuários [Clark et al. 2004]. Para isso, os caminhos da Internet não devem se basear somente em métricas de interesse dos provedores de serviço ou em acordos comerciais, mas também em métricas de interesse dos usuários. Essa possibilidade, porém, esbarra no serviço atualmente utilizado na Internet que é o serviço de "melhor esforço" oferecido pelo IP. Para possibilitar a programação de caminhos a Internet deve suportar qualidade de serviço e a possibilidade dos usuários ou agentes fazerem escolhas

que podem ir contra os acordos dos provedores. Além disso, uma das características fundamentais da Internet é a manutenção da inteligência nas bordas da rede. Essa premissa é baseada no fato que não há ninguém melhor do que os usuários para saber se a aplicação possui bom desempenho ou não [Clark et al. 2004]. Portanto, requisitos como confiabilidade devem ser verificados nas bordas da rede e qualquer outro tipo de verificação no núcleo é considerado redundante. Essa premissa é satisfeita se os caminhos forem escolhidos pelos usuários. Entretanto, a possibilidade do emprego de agentes pode ir em direção oposta a essa premissa já que os agentes podem ser inseridos no núcleo da rede.

A liberdade de escolha de caminhos pelos usuários pode incentivar a disputa e, assim, provocar a redução dos custos de acesso. Uma questão em aberto é a possibilidade da ausência da qualidade de serviço fim-a-fim ser apenas uma consequência da falta de competição entre os ISPs. Caso fosse possível escolher caminhos, usuários que possuíssem aplicações com requisitos especiais poderiam optar por caminhos mais apropriados. Nesse caso, o ISP que entrasse no mercado oferecendo essa possibilidade levaria vantagem sobre os outros que seriam obrigados a modificar os seus serviços para serem competitivos [Clark et al. 2004]. Caso nenhum ISP tomasse a iniciativa, incentivos poderiam ser oferecidos aos ISPs para que esses começassem a oferecer a possibilidade de escolha aos usuários. O problema de oferecer liberdade aos usuários ou agentes é o aumento da complexidade que cada um deve lidar. Por exemplo, os usuários ou agentes precisariam de conhecimento global da rede e ainda poderiam introduzir falhas causadas por más configurações ou más opções. Além dos problemas mencionados, a possibilidade de escolha de caminho pode exigir que cada nó mantenha todos os possíveis caminhos em memória. Esse requisito pode ser mais uma fonte de problemas de escalabilidade.

Escalabilidade dos roteadores

Um dos maiores desafios do aumento acelerado do número de estações e redes na Internet é a escalabilidade. De acordo com um recente relatório do IAB (*Internet Architecture Board*), a escalabilidade é um dos desafios mais críticos em curto prazo para o projeto da Internet do Futuro [Meyer et al. 2007]. Os roteadores são equipamentos que possuem recursos limitados de memória e processamento. Portanto, o aumento indefinido do tamanho das tabelas de roteamento (FIB) e das bases de dados (RIB) pode afetar o desempenho do encaminhamento de pacotes. Cada roteador dispõe de poucos instantes para armazenar um pacote, selecionar a interface de saída baseado no prefixo de rede do endereço de destino e encaminhar para interface correspondente. Além disso, a sobrecarga de controle gerada ao aumentar o número de estações pode se tornar relevante diante da banda passante disponível. Para exemplificar tal crescimento, dados do CIDR (*Classless Inter-Domain Routing*) mostram que desde o início da década de 90 até os dias atuais o número de entradas BGP ativas nas FIBs aumentou de algumas unidades para aproximadamente 310 mil [CIDR 2010].

Os desafios ligados à escalabilidade poderiam ter menor impacto caso o projeto original da Internet fosse seguido. A premissa de organização hierárquica da Internet e a consequente agregação de endereços têm como objetivo tornar o roteamento escalável [Massey et al. 2007, Jen et al. 2008]. Entretanto, as demandas emergentes discutidas nesta seção impedem a agregação de endereços nas FIBs e/ou aumentam a carga de controle da rede. O suporte às redes multidomiciliadas e aos múltiplos caminhos reque-

rem a associação de prefixos distintos a uma mesma rede e requerem a associação de mais de um caminho para um mesmo destino. Já o suporte à mobilidade aumenta a quantidade de informações de controle trocadas para manter uma estação móvel conectada e destrói a premissa de organização hierárquica da Internet. Por fim, a programação de caminhos resulta em armazenamento de informações do roteamento nos usuários. Um efeito indireto que também influencia na escalabilidade é o uso de espaços de endereçamento maiores que o IPv4, como é o caso do IPv6. O uso do IPv6 pode aumentar a oferta de endereços IP, o que resulta em mais estações e redes e mais carga de controle.

Os problemas de escalabilidade já têm sido observados na prática pelo tamanho das tabelas de roteamento BGP. Essas tabelas têm aumentado devido à agregação parcial das faixas de endereços anunciadas pelos ISPs. A Figura 2.5 mostra que o roteador de borda do AS_B anuncia um prefixo de rede que não é agregável ao seu. Já o roteador de borda do AS_C anuncia o seu prefixo desagregado para possibilitar a estação multidomiciliada. Essa agregação parcial é muitas vezes interessante para um provedor de serviço deixar de agregar determinadas faixas de endereços para inclusive balancear carga ou para oferecer serviços diferenciados para os clientes. Dados do CIDR mostram que a razão entre redes que sofreram agregação de prefixos e o total de redes na Internet é atualmente de 38,4% [CIDR 2009], o que demonstra a relevância do problema na Internet. A agregação parcial de endereços, assim como muitos dos desafios da Internet do Futuro, pode culminar no aumento das tabelas de roteamento e no aumento de sobrecarga de controle. Essas consequências representam um entrave ao aumento do número de estações já que a busca por endereços de destino pelos roteadores e o processamento de mensagens podem ser operações custosas computacionalmente.

2.4. Propostas para a Interconexão de Redes

Nesta seção são apresentados novos conceitos e novas propostas que têm por objetivo resolver os desafios destacados na seção anterior. Muitas dessas propostas rompem completamente com a arquitetura original da Internet e algumas delas levam em conta um período de transição para a sua implementação.

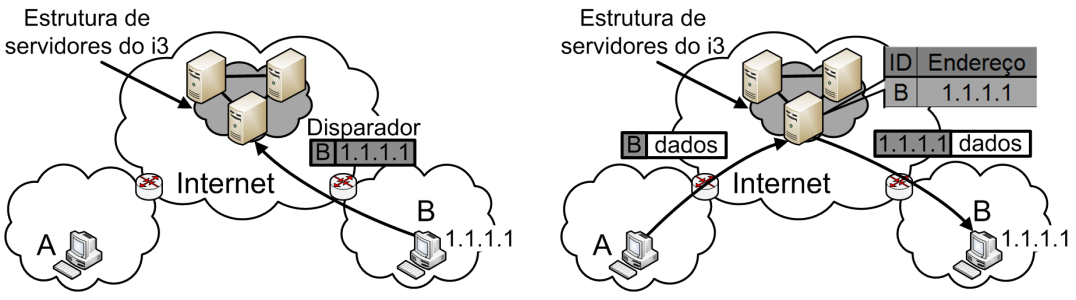
2.4.1. Separação de Localização e Identificação

Uma das principais propostas para minimizar os problemas de escalabilidade devido às demandas por mobilidade e múltiplos domicílios na Internet é a separação da localização física da identificação das estações. Essa separação tem o intuito de quebrar a semântica sobrecarregada do IP e, por isso mesmo, vem sendo considerada como fundamental para a Internet do Futuro [Caesar et al. 2006b]. Esse tipo de abordagem é denominada *Loc/ID split*. O *Loc/ID split* permite que as comunicações realizadas a partir de um identificador invariável se mantenham, mesmo que a localização da estação mude. Ainda que em outro nível, o procedimento é semelhante ao DNS que separa a identificação do nome de um sítio da Internet do seu endereço IP. Entretanto, assim como no caso do DNS, é também necessário algum sistema de mapeamento para associar identificador e localizador. O mapeamento dinâmico e escalável entre identificador e localizador é um problema de pesquisa em aberto [Iannone e Bonaventure 2007, Luo et al. 2009].

O i3 (*Internet Indirection Infrastructure*) [Stoica et al. 2004] baseou sua proposta

na constatação de que serviços como mobilidade e multicast endereçam estações de maneira indireta para funcionarem na Internet atual. Por exemplo, para prover suporte à mobilidade há possivelmente o emprego de agentes domiciliares assim como para prover suporte ao multicast há o emprego de endereço de grupo. Em ambos os casos o endereço de destino utilizado pela fonte não é o endereço da estação final. O i3 utiliza essa constatação para propor uma estrutura sobrecamada para oferecer suporte unificado a qualquer tipo de serviço que possa usufruir de endereçamento indireto.

O modelo de serviço do i3 basicamente desassocia o ato de enviar pacotes do ato de recebê-los. Para isso, cada fonte envia os pacotes para o identificador do destino, ao invés de enviar para o endereço. O destino dos pacotes demonstra interesse nesses pacotes enviando disparadores (*triggers*) contendo o seu identificador e o seu endereço. Tanto os pacotes quanto os disparadores são enviados para a estrutura de servidores do i3. Em tal estrutura, os pacotes recebidos são associados aos disparadores correspondentes para que haja o mapeamento do identificador do pacote no endereço de destino. Na estrutura i3, cada servidor é responsável por um conjunto de identificadores. Esse servidor mantém o mapeamento dos identificadores que estão sob sua responsabilidade. Logo, todo pacote recebido na estrutura é encaminhado para o servidor responsável para que ele realize o mapeamento e encaminhe o pacote para o endereço do destino correspondente. O mapeamento é mantido no servidor da estrutura i3 de maneira volátil. Assim, de tempos em tempos os destinos devem atualizar o seu mapeamento na estrutura. A Figura 2.7 ilustra todo procedimento de encaminhamento de pacotes no i3. A Figura 2.7(a) mostra o envio do disparador realizado pela estação de destino B para demonstrar interesse em um determinado conteúdo. O disparador contém o seu identificador e o seu endereço. A estrutura i3, por conseguinte, armazena o interesse. Assim, sempre que a estrutura do i3 recebe pacotes destinados ao identificador da estação B, a estrutura já sabe para que destino encaminhar, como visto na Figura 2.7(b). O endereçamento indireto proposto pelo i3 desassocia as fontes dos destinos. Assim, as fontes não precisam conhecer nem o número de destinos nem os endereços para os quais os seus pacotes são enviados, e da mesma maneira, os destinos não precisam conhecer nem o número de fontes nem os seus endereços. Uma importante característica do i3 é que um determinado identificador pode estar associado a mais de um endereço de destino. Nesse caso, o mapeamento ocorre para todos os destinos cujo identificador é igual a um número mínimo de bits do identificador mantido na estrutura de servidores do i3.



(a) Envio do disparador. (b) Encaminhamento de pacotes.

Figura 2.7. Encaminhamento de pacotes realizado pelo i3.

No caso dos nós móveis, a conexão pode ser mantida visto que ela é realizada utilizando o identificador das estações de origem e destino e não os endereços físicos. Entretanto, o nó móvel ao mudar de rede de acesso, muda de endereço físico e, portanto, deve atualizar o mapeamento na estrutura i3. Uma vez que as fontes não precisam conhecer o endereço do destino, a conexão é mantida. O i3 requer a atualização do mapeamento nos servidores, o que pode implicar problemas de latência de atualização, além de problemas de escalabilidade. Outro ponto é o aumento dos caminhos seguidos pelos pacotes em consequência do encaminhamento indireto.

O LISP (*Locator/Id Split Protocol*) [Farinacci et al. 2009b, Saucez et al. 2009] é um protocolo desenvolvido pela CISCO para lidar com os problemas dos múltiplos domicílios na Internet. O LISP é uma proposta que pode ser implementada de maneira gradual, o que representa uma vantagem. O LISP também realiza endereçamento indireto assim como o i3. Para isso, ele divide o espaço de endereçamento no espaço de endereçamento local, composto pelas redes de borda da Internet, e no espaço de endereçamento interdomínio, composto pelos roteadores da DFZ. Os roteadores dos ASes de borda da Internet possuem interfaces conectadas à DFZ configuradas com endereços denominados *Routing Locators* (RLOCs). Esses endereços são conhecidos por todos os roteadores interdomínio. Já os endereços das estações da rede local, denominados *Endpoint IDentifiers* (EIDs), possuem escopo limitado à rede de acesso. Portanto, os endereços das estações não são conhecidos pelos roteadores interdomínio e os RLOCs não podem identificar estações. O encaminhamento de pacotes fim-a-fim requer, portanto, o uso dos dois tipos de endereços. Nessa direção, os EIDs de uma rede local são associados aos RLOCs dos ISPs pelos quais possuem acesso à Internet. Um EID pode estar associado a mais de um RLOC, o que permite os múltiplos domicílios. Equivalentemente, um RLOC pode estar associado a um prefixo de endereços locais. Outra vantagem do LISP é que ele dispensa o uso de estrutura de servidores, como usado no i3. A Figura 2.8 ilustra os endereços utilizados pelo LISP. Na figura, a estação EID_A possui associado o endereço RLOC_A na DFZ. Já o EID_B possui dois endereços RLOC associados, RLOC_{B1} e RLOC_{B2}, para múltiplos domicílios.

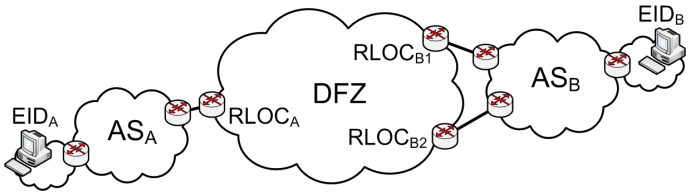
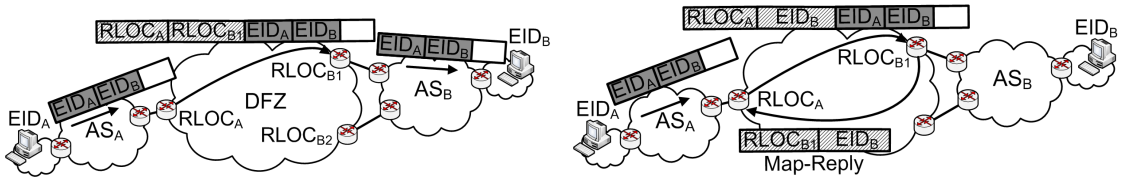


Figura 2.8. Endereçamento no LISP.

No LISP, os pacotes são encaminhados na DFZ utilizando como endereços de origem e destino os endereços dos RLOCs já que os endereços locais não são conhecidos. Entretanto, entre a estação de origem do pacote e o seu roteador de borda e entre o roteador de borda no destino e a estação de destino, os endereços usados são os EIDs de origem e destino. Para que os dois tipos de endereços sejam usados pelos pacotes, é necessário que haja o mapeamento dos EIDs nos endereços dos RLOCs. Após o mapeamento, os pacotes são encapsulados através de túneis entre os roteadores de borda. Tal procedimento

é conhecido por mapeamento e encapsulamento (*Map & Encap*). Os roteadores de borda realizam o *Map & Encap* já que eles são os únicos que conhecem a associação entre os EIDs e os RLOCs. Essa característica é uma vantagem do LISP, pois torna o seu funcionamento transparente para os outros nós da rede, além de dispensar modificações na estrutura da Internet. A Figura 2.9(a) ilustra os cabeçalhos utilizados por um pacote no LISP em cada um dos trechos percorridos: EID_A de origem até RLOC_A, RLOC_A até RLOC_{B1} e RLOC_{B1} até o destino EID_B. O mapeamento no LISP é realizado por dois sistemas, o de base de dados e o de memória cache. O de base de dados seleciona o RLOC que é usado como endereço de origem dos pacotes sendo enviados de uma rede local. Além disso, o sistema de base de dados decide se um pacote recebido da Internet deve ser desencapsulado ou não. O sistema de memória cache, por outro lado, é usado para selecionar os RLOCs das redes de acesso da estação de destino. Para tal, o mapeamento EID de destino e RLOC de destino são também armazenados nesse cache. Caso haja uma busca mal-sucedida à cache sobre o mapeamento de uma determinada estação de destino, o sistema de mapeamento do LISP faz uma requisição a um sistema de mapeamento através do protocolo de mapeamento distribuído (*Mapping Distribution Protocol*).



(a) Encaminhamento direto. (b) Uso do *Map-Reply*.

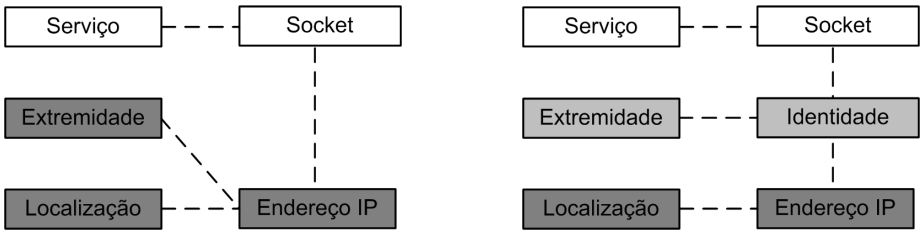
Figura 2.9. Encaminhamento de pacotes no Lisp.

No LISP, o encaminhamento de pacotes assume que a estação de origem conhece o identificador do destino e o seu próprio endereço. A resolução do identificador do destino no EID é realizada através de um sistema de resolução de nomes como o DNS. O pacote é então encaminhado pela estação de origem até o roteador de borda através de uma rota *default*. Ao chegar no roteador de borda, ele deve descobrir o mapeamento do EID do destino no RLOC correspondente. Esse procedimento depende da versão do LISP. As primeiras versões do LISP, LISP 1 e 1.5, assumem que o endereço EID é roteável na Internet. Assim, o roteador de borda da origem encapsula o pacote com um cabeçalho definido pelo LISP com o seu endereço como endereço de origem e o endereço da estação de destino como endereço de destino. Esse pacote é encaminhado pela DFZ até que chegue ao roteador de borda que conhece o mapeamento EID/RLOC do destino. O roteador de borda do destino envia uma mensagem ao roteador de borda da origem (*Map-Reply*) informando o mapeamento, como visto na Figura 2.9(b). O roteador de borda da origem armazena essa informação em sua memória cache assim que a recebe. Da mesma maneira, o roteador de borda do destino armazena o mapeamento em sua memória cache para evitar uma possível busca subsequente. As versões 2 e 3 do LISP não consideram mais que os EIDs são roteáveis na Internet. Assim, para conhecer o mapeamento EID/RLOC, o LISP utiliza um protocolo de distribuição de mapas (*Mapping Distribution Protocol*) para fazer requisições explícitas (*Map-Request*) a um serviço de mapeamento distribuído. Tais sistemas podem ser implementados baseados em sistemas de DNS, LISP versão 2; ou

em sistemas baseados em DHTs (*Distributed Hash Table*), LISP versão 3. Em ambos os casos, os sistemas respondem (*Map-Reply*) apenas às requisições recebidas.

A separação da localização da identificação proposta no LISP facilita o uso de estações multidomiciliadas visto que desassocia o endereço da estação do endereço de seu ISP. Entretanto, essa separação resulta também no emprego de sistemas de mapeamento, que introduzem um atraso para resolver EIDs em RLOCs. Como esse atraso pode ser grande, ele ainda representa um entrave à mobilidade das estações e de redes embora a separação da localização da identificação dos nós beneficie a mobilidade em um primeiro momento. O LISP possui como ponto de estudo futuro encontrar maneiras mais eficientes para lidar com a mobilidade de estações e de redes.

O HIP (*Host Identity Protocol*) [Moskowitz et al. 2008] é outro protocolo para *Loc/ID split* [OpenHIP 2009]. Para isso, ele propõe uma nova camada entre a camada de redes e a camada de transporte. Assim, a camada de transporte interage com a camada de identificação e a camada de identificação interage com a camada de rede. O HIP utiliza os conceitos de identidade e identificador. A identidade de um nó se refere à entidade abstrata que é identificada. Já o identificador se refere ao padrão binário que é utilizado no processo de identificação. Por exemplo, a identidade é o nome de uma estação (`rio.gta.ufrj.br`) e o identificador uma chave pública criptográfica. O HIP utiliza chave pública como identificador, embora qualquer tipo de identificador possa ser usado. A vantagem de utilizar uma chave pública como identificador é a possibilidade de autenticar a origem dos pacotes recebidos. Como consequência, pode-se evitar a violação dos pacotes em trânsito como feito em ataques do tipo homem-no-meio (*man-in-the-middle*). O HIP considera que identificadores que não utilizem criptografia devem ser apenas utilizados em ambientes considerados seguros. A Figura 2.10 ilustra a arquitetura atual para identificação e localização que sobrecarrega o endereçamento IP e a arquitetura proposta pelo HIP que separa a localização da identificação de uma estação.



(a) Arquietura atual da Internet. (b) Arquitetura do HIP.

Figura 2.10. Arquiteturas de identificação e localização.

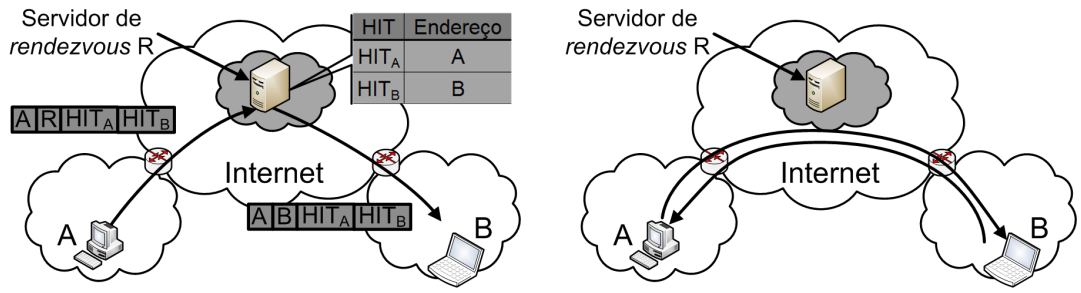
Diferente da identidade de um nó no HIP que é uma abstração, o identificador pode ser utilizado nas comunicações. Entretanto, devido ao tamanho e a variabilidade possível, a maneira escolhida pelo HIP para representar o identificador nos pacotes do protocolo é um resumo obtido através do uso de uma função *hash*. Dessa forma, a representação do identificador torna o seu uso mais simples já que possui tamanho fixo, além de significar um formato consistente independente dos algoritmos criptográficos empregados. O resultado da função *hash* é denominado como rótulo da estação (*Host Identity Tag* - HIT) e, consequentemente, deve ser único globalmente.

Uma comunicação com HIP é iniciada a partir de um processo de autenticação e troca de chaves. Portanto, sempre antes de enviar dados é realizada uma troca de base através do BEX (*Base Exchange Protocol*) e uma associação HIP se torna ativa. Esse protocolo é executado fim-a-fim, ou seja, entre as estações finais. O procedimento de autenticação envolve a troca de desafios criptográficos para autenticação. Todo o procedimento pode ser atualizado quando as chaves criptográficas expiram ou quando uma estação se desloca.

A camada de transporte estabelece conexões utilizando os HITs de origem e destino que são oferecidos pela camada introduzida pelo HIP para identificação das estações envolvidas na comunicação. Entretanto, para os pacotes serem encaminhados na Internet, a camada introduzida pelo HIP precisa substituir os HITs pelos endereços IP correspondentes. Para isso, a estação de origem dos pacotes substitui o HIT de origem pelo endereço IP correspondente e obtém via um sistema de resolução de nomes, o endereço IP do destino. No HIP, a resolução de nomes não retorna apenas o endereço IP da estação, mas também informações como o HIT e a chave pública. Após a resolução de nomes, a comunicação de dados inicia-se entre a estação de origem e destino. Uma vez descoberto o endereço de um nó, esse endereço é utilizado até que a comunicação se encerre ou que o nó de destino se desloque. Essa característica difere o HIP do LISP, já que no LISP o encaminhamento dos pacotes sempre utiliza túneis. No HIP, se uma estação se mover, a própria estação pode avisar à estação de destino qual o seu novo endereço físico. Entretanto, se uma nova estação desejar estabelecer uma conexão com a estação que se moveu, a resolução do nome irá retornar o endereço IP antigo já que o mapeamento no DNS não foi atualizado. Para esses casos, o HIP usa servidores na Internet para evitar que as estações móveis precisem sempre atualizar o DNS.

O HIP define um servidor estático chamado de *rendezvous* para auxiliar em um processo de encaminhamento indireto, semelhante ao usado no i3. Para isso, o nome da estação móvel é mapeado pelo DNS no endereço IP do *rendezvous*. O servidor *rendezvous* possui atualizada a localização da estação móvel. Parte-se da premissa que é melhor atualizar o endereço IP em um servidor na Internet que o DNS. Logo, uma estação da Internet que deseja se comunicar com uma estação móvel realiza a resolução de nomes e recebe o endereço IP do servidor de *rendezvous* e o HIT da estação móvel. A estação da Internet inicia o processo de autenticação e troca de chaves utilizando o endereço IP e o HIT recebido. A mensagem é encaminhada até o servidor de *rendezvous* que ao perceber que o HIT da mensagem não é um dos seus, verifica em seus registros se possui o mapeamento correspondente. Todas as estações que desejam utilizar um servidor de *rendezvous* devem se registrar nesses servidores. Portanto, caso a mensagem seja destinada a um HIT cujo mapeamento no endereço IP é conhecido, o servidor encaminha o pacote até a estação móvel correspondente. Ao receber a primeira mensagem de autenticação e troca de chaves, a estação móvel responde diretamente à estação de origem e o restante da comunicação passa a ser direta desde então sem o intermédio do *rendezvous*. Se a estação móvel mudar de rede novamente, ela atualiza a outra estação e o servidor *rendezvous* da sua nova localização. A Figura 2.11 ilustra o funcionamento do HIP com o uso do servidor de *rendezvous*. Na Figura 2.11(a), a estação de origem A envia a primeira mensagem com o endereço IP do servidor de *rendezvous* R obtido após a resolução do nome no DNS. O servidor de *rendezvous*, então, encaminha a mensagem até a estação

móvel. Em seguida, a estação móvel se comunica diretamente com a estação que originou a comunicação, como visto na Figura 2.11(b).



(a) Encaminhamento através do *rendezvous*. (b) Encaminhamento direto.

Figura 2.11. Encaminhamento de pacotes no HIP.

Apesar do HIP encaminhar pacotes de uma maneira diferente do LISP, ele também pode sofrer problemas com a mobilidade de estações devido à latência de atualização do mapeamento. Em redes móveis em velocidades altas, essa latência pode não ser suficiente para atender as demandas de serviços com restrições de atraso. O HIP provê ainda suporte a múltiplos domicílios, já que mais de um endereço pode estar relacionado à mesma identidade [Nikander et al. 2008].

O Shim6 [Nordmark e Bagnulo 2009] é uma proposta para proporcionar tolerância a falhas a estações multidomiciliadas em domínios IPv6. O Shim6 mantém comunicações correntes mesmo em caso de falha de alguns localizadores. Caso uma parte das conexões de uma rede, as outras ativas devem manter a comunicação. No pior caso, se todas as conexões falharem, o tempo de restabelecimento da comunicação é igual ao tempo de atualização do DNS e das atualizações se propagarem pela rede. Nesse sentido, o Shim6 possibilita o uso dos múltiplos domicílios. Semelhante ao HIP, o Shim6 atribui um nome especial para o identificador do nó, chamado de ULID (*Upper Layer IDentifier*). Já o localizador da estação não possui nome especial e se refere ao endereço IPv6 da camada de rede. Diferente das propostas anteriores, entretanto, o endereço IPv6 utilizado no primeiro contanto com a outra estação cumpre também o papel de ULID. Logo, o Shim6 não propõe o uso de um nome espaço de identificação. Porém, caso o localizador mude ao longo da comunicação entre os nós, o ULID permanece constante. Essa possibilidade é considerada pouco frequente visto que o Shim6 não foi desenvolvido diretamente para redes cujos nós mudem de localização dinamicamente, como seria o caso das redes móveis.

O Shim6 é representado como uma camada diretamente acima da camada de rede. Assim, tanto as aplicações quanto os protocolos de camadas acima da camada Shim6 utilizam os ULIDs que foram mapeados a partir dos localizadores das estações de origem e destino. A camada Shim6 mantém o mapeamento por par de ULIDs. Portanto, todas as conexões envolvendo o mesmo par origem-destino compartilham o mesmo mapeamento. O mapeamento deve ser realizado de maneira consistente na origem e no destino para que os protocolos de camadas superiores vejam os pacotes sendo enviados utilizando os ULIDs fim-a-fim. Essa característica é mantida mesmo que os pacotes sejam encaminhados pela rede utilizando os localizadores como endereços IP e mesmo que os localiza-

dores mudem. Além disso, o mapeamento realizado para o par de nós deve ser mantido independentemente do protocolo de camada superior e de qualquer outra conexão estabelecida. Os protocolos de camadas superiores que estão cientes da operação do Shim6 podem associar mais de um par de localizadores a um único par de ULIDs.

No Shim6, uma determinada estação de origem inicia uma transmissão caso alguma de suas aplicações possua pacotes a enviar. Nesse momento, o Shim6 pode se comunicar com o destino do pacote para conhecer possíveis pares alternativos de localizadores associados ao par de ULIDs. Essa comunicação é possível visto que o ULID é igual ao localizador da estação, o que dispensa o uso de sistemas de mapeamentos. Os múltiplos pares de localizadores são utilizados para casos de falha. Se uma comunicação falhar, o Shim6 pode testar novos pares de localizadores até que um deles restabeleça a comunicação. Ao encontrar esse par, os localizadores são utilizados nos próximos pacotes, mas uma extensão de cabeçalho é utilizada para que o destino saiba qual o ULID correspondente ao novo par de localizadores. Essa extensão é chamada de rótulo de contexto (*Context Tag*). O uso do rótulo de contexto permite que as mudanças dos localizadores permaneçam transparentes para os protocolos de camadas superiores.

O fato de o localizador inicial ser igual ao identificador pode levar a períodos de instabilidade. Caso o localizador de uma estação mude, nada impede que o localizador anterior seja atribuído a outro nó da rede. Assim, há a possibilidade de duas estações diferentes utilizarem o mesmo identificador já que o identificador é inicialmente igual ao localizador. O Shim6 evita essa possibilidade forçando que as comunicações sejam terminadas quando uma ULID se tornar inválida. Um mecanismo de recuperação de contexto é então acionado para que a outra estação na comunicação fique ciente que a ULID não está mais relacionada com o localizador anterior. Esse procedimento é uma consequência de se evitar sistemas de mapeamento, como p. ex. o *rendezvous* do HIP.

O mapeamento entre identificadores e localizadores ou entre endereços locais e endereços intradomínio é um problema em aberto para os diferentes protocolos de *Loc/ID Split*. No LISP, por exemplo, muitos sistemas são avaliados para tornarem o sistema de mapeamento escalável e seguro. Os sistemas de mapeamento podem ser classificados em PUSH ou PULL. Nos sistemas PUSH, o mapeamento mantido por elementos centralizadores é enviado aos roteadores sem que os roteadores façam requisições. Essa estratégia pode aumentar a carga de controle enviada na rede, além de manter todos os mapeamentos conhecidos pelos roteadores de borda. O ponto positivo é que o mapeamento não sofre atrasos. Propostas como o NERD (*Not-so-novel EID to RLOC Database*) [Lear 2010] anunciam os mapeamentos conhecidos a todos os roteadores da rede sem a necessidade de requisições. Os sistemas PULL funcionam através de requisições. Portanto, sempre que um roteador de borda não conhecer um determinado mapeamento, ele envia uma requisição para o sistema de mapeamento. Tais sistemas de mapeamento podem funcionar como um DNS ou uma DHT (*Distributed Hash Table*) [Iannone e Bonaventure 2007]. Iannone e Bonaventure demonstram que sistemas de mapeamento baseados em DNS não sofrem com problemas de escala [Iannone e Bonaventure 2007]. Para isso, os autores demonstram que é possível controlar o tamanho da memória cache dos nós a partir do ajuste do tempo de permanência dos mapeamentos na memória. Luo *et al.* [Luo et al. 2009] demonstram que sistemas baseados em DHTs também podem alcançar desempenho satisfatório em termos de escalabilidade. Sistemas como o LISP-ALT [Farinacci et al. 2009a]

e LISP-CONS [Brim et al. 2008] são sistemas híbridos, ou seja, que utilizam tanto a estratégia PULL quanto a PUSH. O LISP-ALT define uma arquitetura na qual alguns roteadores são responsáveis por manter os mapeamentos. Esses roteadores trocam informações baseado na estratégia PUSH. Já os roteadores de borda requisitam o mapeamento aos roteadores responsáveis. O LISP-CONS difere do LISP-ALT pela organização dos roteadores responsáveis por manter os mapeamentos. No LISP-CONS, a estrutura desses roteadores é hierárquica. Portanto, os mapeamentos são trocados entre os roteadores em nível hierarquicamente superior através da estratégia PUSH. Já os roteadores de borda requisitam os mapeamentos aos roteadores de nível hierárquico inferior que consultam os roteadores em nível superior também através de requisições.

2.4.2. Roteamento plano

O roteamento plano é outra possibilidade para contornar os problemas oriundos da semântica sobrecarregada do endereço IP. Para isso, o roteamento plano extrapola o conceito da separação entre localização e identificação removendo o conceito de localização. O objetivo é rotear pacotes baseado apenas na identificação dos nós. Assim, torna-se necessário garantir a unicidade do identificador e não mais a relação entre a localização do nó e a topologia de rede. Uma vez que o roteamento seja baseado no identificador, e não mais em um endereço correlacionado com a topologia de rede, esse nó pode se deslocar sem prejuízo das suas conexões. Além disso, não realizar a resolução de nomes evita o emprego de infraestrutura de rede, p. ex. DNS, e aumenta a disponibilidade da comunicação já que o sistema de resolução de nomes não é mais um ponto de falha. O roteamento plano deve ser investigado apesar da premissa da Internet de manter no cabeçalho dos pacotes informação estruturada da localização das estações. Entretanto, trabalhos da área demonstram que esse tipo de proposta ainda enfrenta problemas de escalabilidade.

Até o momento, o ROFL (*Routing On Flat Labels*) [Caesar et al. 2006b] é o principal trabalho em roteamento plano. O ROFL utiliza rótulos para identificação dos nós e desassocia a identificação da localização na rede. A operação do ROFL é baseada no conceito de DHTs (*Distributed Hash Tables*), utilizadas comumente por protocolos de redes par-a-par. A motivação principal do uso das DHTs é a sua capacidade de distribuição homogênea de carga ou funcionalidades. Em uma rede par-a-par, p. ex., é desejável que os dados armazenados pelos nós da rede sejam distribuídos homogeneamente para evitar nós sobrecarregados. A característica de distribuição homogênea, por conseguinte, leva as DHTs a serem aplicadas em sistemas que evitam estruturas hierárquicas. Por isso, o ROFL utiliza as DHTs no roteamento ao invés de distribuição de conteúdo.

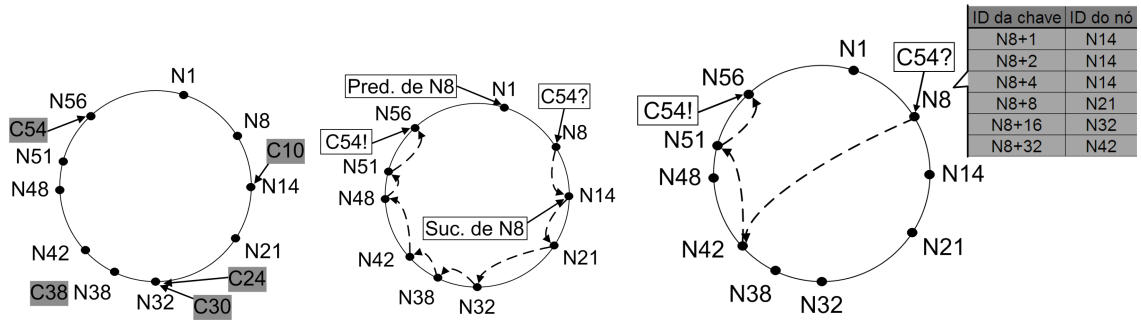
As propostas para roteamento plano, incluindo o ROFL, devem funcionar tanto para comunicações entre nós no mesmo AS quanto para nós em ASes diferentes. Portanto, as propostas devem definir protocolos para o estabelecimento e a manutenção das comunicações em ambos os casos. Os protocolos propostos pelo ROFL são baseados no Chord [Stoica et al. 2003] e no Canon [Ganesan et al. 2004], que são sistemas de distribuição que utilizam DHTs. O primeiro é utilizado pelo ROFL como base para as comunicações intradomínio. Já o segundo, é utilizado como base para as comunicações interdomínio. Antes de apresentar o funcionamento do ROFL é interessante apresentar o funcionamento do Chord e do Canon.

O Chord é um protocolo originalmente proposto para busca de conteúdo em redes par-a-par. No Chord, cada tipo de dado é associado a uma chave que é escolhida em função do identificador utilizado na camada de aplicação, p. ex. um nome de um arquivo em um sistema par-a-par. Já na camada de rede, o Chord associa um identificador a cada uma das chaves e também associa identificadores a cada um dos nós. Ambos os identificadores são obtidos com o uso de funções *hash*. Por fim, o Chord mapeia um identificador de chave a um nó da rede. Assim, cada nó fica responsável por uma ou mais chaves. Caso a chave identifique um dado, este pode ficar armazenado no nó correspondente. As principais vantagens do Chord são a distribuição uniforme de chaves que evita a sobrecarga de algum nó em particular e o armazenamento de informações locais de roteamento em cada nó. O objetivo é aumentar a escalabilidade do sistema ao dividir a carga de armazenamento de dados e de informação de controle.

O mapeamento entre os identificadores das chaves e dos nós é realizado em um espaço de identificação circular de módulo 2^m, onde m é o tamanho do identificador em bits. Assim, um identificador de chave `C` é mapeado no primeiro nó cujo identificador é igual a `C` ou, se o nó com identificador igual não existir, ao primeiro nó encontrado seguindo o espaço de identificação no sentido horário. A Figura 2.12(a) ilustra um espaço de identificação de módulo 2^6. Na figura, os identificadores que começam com `C` são os identificadores das chaves e os que começam com `N`, os dos nós. A figura mostra que o identificador da chave `C10` não foi diretamente mapeado em um nó com um identificador de mesmo número. Logo, `C10` foi mapeada no próximo nó no sentido horário, ou seja, no nó `N14`. Já o nó `N38` possui um identificador de chave que coincide com o seu próprio identificador. O processo de localização básico de um identificador de chave consiste em uma busca no espaço circular. Essa busca é realizada salto-a-salto, no qual cada salto é um identificador de nó. A localização termina quando o identificador da chave pertencer ao intervalo entre o identificador do nó analisado e o seu próximo nó no sentido horário. Nesse caso, a busca retorna o identificador desse próximo nó. No Chord, cada nó armazena o identificador do próximo nó no sentido horário e do anterior. Esses nós são chamados de nó sucessor e predecessor, respectivamente. A Figura 2.12(b) mostra como é realizado o processo de localização de um identificador de chave no Chord a partir do nó `N8`. Na figura, os nós `N1` e `N14` são os nós sucessor e predecessor do nó `N8`, respectivamente. O nó `N8` busca a localização do identificador da chave `C54`. O processo de localização é repetido pelos nós no círculo até encontrar a chave desejada.

O espaço de identificação circular permite que o nó responsável por um determinado dado seja sempre encontrado, já que a busca pode apenas ser realizada no sentido horário. Na Internet, entretanto, a rede é organizada hierarquicamente e a busca por um determinado nó pode exigir mais estados por roteador. Enquanto no Chord cada nó precisa apenas conhecer o seu sucessor e predecessor no espaço circular, na Internet, cada nó precisa saber o próximo salto para cada possível destino. A vantagem da redução de estados proposta pelo Chord, entretanto, insere um problema. Quanto menos estados forem armazenados por nó, maior é a quantidade de saltos que o procedimento de localização deve realizar. Esse compromisso é abordado pelo Chord ao introduzir uma tabela de roteamento por nó. No Chord, cada nó mantém uma tabela de m linhas para agilizar o processo de localização dos identificadores das chaves. Cada linha da tabela identifica o nó responsável pelos identificadores de chaves distantes até 2^m. Portanto, cada linha

da tabela é uma tupla composta pelo identificador da chave (igual ao número do identificador do nó realizando a busca mais 2^{i-1}, onde $1 \leq i \leq m$) e pelo identificador do nó responsável por essa chave. Caso o identificador da chave seja maior que o último identificador na lista, a busca é adiantada até o primeiro nó no sentido horário após a lista. A Figura 2.12(c) ilustra o procedimento de localização de `C54` realizado por `N8` com o uso de tabela de roteamento. O nó `8` adianta a sua busca até `N42` já que o identificador da chave buscada é maior que o identificador do último nó da tabela. O nó `42` também adianta a localização saltando `N48`. Ao combinar o espaço de identificação circular com a tabela de roteamento, o Chord propõe uma solução para localização de identificadores de nós que aumenta a escalabilidade do sistema. A ideia de organizar a rede em um espaço de identificação circular pode ser aplicada à Internet tomando-se o cuidado para que o requisito de unicidade de identificação seja garantido. Uma vez que um nó tenha um identificador único associado, independente da sua localização, seu identificador permanece o mesmo e a sua posição no espaço de identificação também. Isso permite que os nós possam se deslocar sem quebrar as conexões estabelecidas.

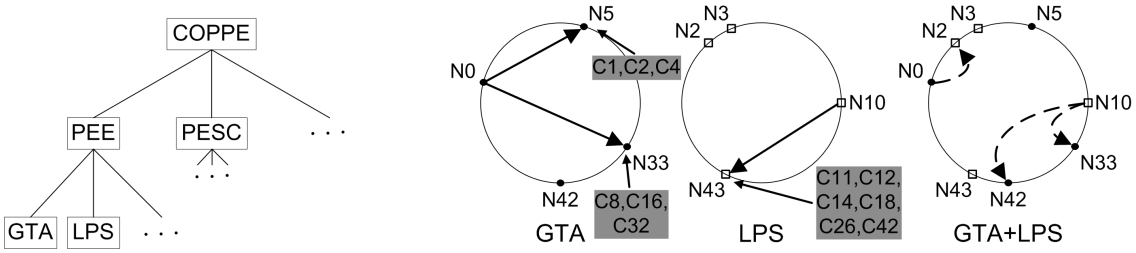


(a) Mapeamento entre identificadores de chave e nó. (b) Busca de identificador no Chord. (c) Busca de identificador no Chord usando tabelas.

Figura 2.12. Funcionamento do Chord.

Um problema do uso das DHTs como no Chord é que, embora o seu uso possibilite a distribuição balanceada de carga entre os nós, o projeto de sistemas escaláveis está normalmente associado a sistemas com estruturas hierárquicas. Portanto, uma maneira de aumentar a escalabilidade do sistema é aplicar hierarquia sem perder as principais vantagens das DHTs. O Canon [Ganesan et al. 2004] é uma proposta para estender o Chord adicionando hierarquia à sua estrutura. Embora contraditório, o objetivo é construir um sistema híbrido para reunir as principais vantagens das DHTs e dos sistemas com estruturas hierárquicas. Uma das motivações é a adaptação mais suave à realidade da Internet que é organizada de maneira hierárquica e dividida em diferentes ASes. A Figura 2.13(a) mostra um exemplo ilustrativo da estrutura física da rede da COPPE, que é a instituição responsável pela pós-graduação em engenharia da UFRJ. A COPPE é dividida em diferentes programas (Engenharia Elétrica, Mecânica etc.) e cada programa possui laboratórios afiliados. Portanto, a organização das redes segue a mesma estrutura hierárquica como na Internet, onde cada uma das redes representa um domínio. A estrutura utilizada é uma árvore na qual os laboratórios GTA e LPS são folhas e a raiz é a COPPE.

No Canon, cada domínio folha na árvore define um espaço de identificação circular, assim como no Chord. A diferença está nos domínios internos da árvore que definem



(a) Hierarquia entre domínios. (b) União de espaços de endereçamento.

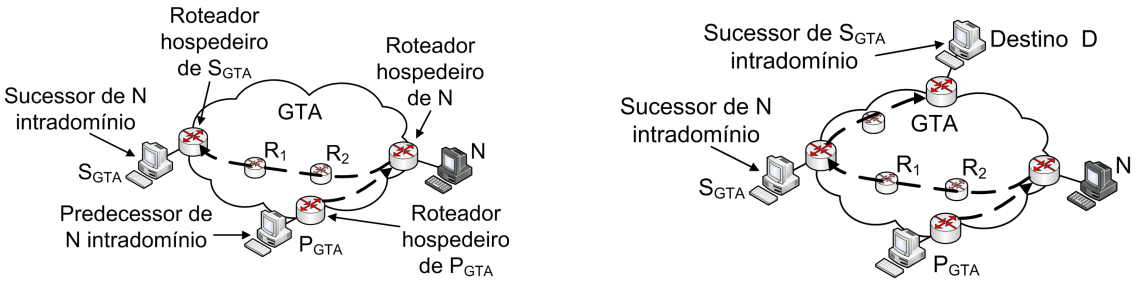
Figura 2.13. Funcionamento do Canon.

um espaço de identificação circular que une os seus nós e mais os nós dos seus domínios filhos. Essa estrutura é repetida hierarquicamente até alcançar o domínio raiz, no exemplo da Figura 2.13(a), o domínio da COPPE. A identificação dos nós deve ser única globalmente e cada nó deve criar uma tabela de roteamento baseada no módulo do espaço de identificação circular da rede. A Figura 2.13(b) ilustra o espaço de identificação circular módulo 2^6. O nó `N0` do GTA possui entradas em sua tabela de roteamento para identificadores de chaves que estão associadas aos nós `N5` (chaves 1, 2 e 4) e `N33` (chaves 8, 16 e 32). Já o nó `N10` do LPS possui entradas para identificadores de chaves associadas ao nó `N43` (chaves 11, 12, 14, 18, 26 e 42). A união do espaço de endereçamento do GTA com o do LPS mantém todas as entradas estabelecidas. Com a união, novas entradas são adicionadas seguindo duas regras pré-definidas. A primeira regra define que um nó `N` do GTA pode criar uma ligação com um nó `N'` do LPS se e somente se `N'` for o nó mais próximo que esteja distante de no máximo 2^m. Já a segunda regra define que o nó `N'` que atende o requisito da regra anterior deve ser mais próximo de `N` do que qualquer outro vizinho de `N` do GTA. Essa segunda condição limita o número de entradas aos nós do espaço de endereçamento vizinho mais próximos dos identificadores de chaves entre $\mathtt{N} < \mathtt{N} + 2^{i-1} < \mathtt{N} + 2^m$ do que os nós do próprio espaço. Na Figura 2.13(b), os espaços de endereçamento unidos demonstram as entradas da tabela de roteamento que foram formadas pelos nós `N0` e `N10` com o processo de união de espaços. Note que `N0` pode manter entradas para `N2` devido às distâncias 1 e 2 e para `N10` devido à distância 8, de acordo com a primeira regra. Entretanto, `N10` está mais distante de `N0` que `N5` que pertence também ao GTA. Portanto, a entrada correspondente não é adicionada por `N0`, como definido pela segunda regra.

A união de espaços circulares é feita recursivamente desde os nós folhas até o nó raiz. Ao final, o espaço de endereçamento circular é formado englobando todos os nós da rede. O roteamento realizado no Canon segue uma abordagem gulosa na qual um determinado pacote é encaminhado até o identificador no mesmo nível hierárquico mais próximo ao destino. Por exemplo, na Figura 2.13(b), os pacotes enviados pelo nó `N2` ao nó `N42` seguem o caminho no LPS até `N10` e de `N10` até `N42`. É importante mencionar que as entradas existentes na tabela de roteamento de cada nó são preservadas mesmo após o processo de união. Além disso, o processo realizado pelo Canon pode levar em conta domínios que pertençam a ASes diferentes.

Semelhante ao Chord, cada nó no ROFL possui um nó sucessor e um nó predecessor na rede intradomínio. Entretanto, o ROFL considera que os nós pertencem a

uma rede local de acesso que oferece conectividade via um roteador de interconexão, denominado pelo ROFL como roteador hospedeiro. Esse roteador armazena os caminhos completos que o ligam até os roteadores hospedeiros dos nós sucessores e predecessores dos seus nós hóspedes. Da mesma maneira, os roteadores hospedeiros dos nós sucessores e predecessores também armazenam o caminho completo no sentido reverso. O caminho completo é definido como a sequência salto-a-salto dos identificadores de todos os roteadores físicos entre hospedeiros e é utilizado durante o encaminhamento de pacotes para roteamento pela fonte. A Figura 2.14(a) ilustra os nós sucessor ($\texttt{S}_{GTA}$) e predecessor ($\texttt{P}_{GTA}$) de `N` no domínio do GTA. Note que tanto o caminho armazenado até o nó sucessor (N, R_1, R_2, S_{GTA}) quanto o caminho até o nó predecessor (N, P_{GTA}) iniciam e terminam nos roteadores hospedeiros. A rede também pode possuir roteadores que originam ou recebem tráfego. Nesse caso, os roteadores também possuem sucessores e predecessores e os caminhos são armazenados semelhantemente ao caso dos nós hóspedes. O roteador hospedeiro é responsável por interconectar um nó que entra na rede ao espaço de identificação circular. Para tanto, o roteador hospedeiro primeiramente autentica o nó entrante, para somente após encontrar os roteadores sucessores e predecessores desse nó no espaço circular. Caso o roteador hospedeiro falhe, os nós hóspedes devem perceber isso através de temporizadores expirados. Nesse caso, os nós hóspedes procuram um roteador alternativo para entrar novamente na rede ou elegem um roteador hospedeiro de reserva. Essa última opção só é possível se mais de um roteador hospedeiro for escolhido desde a primeira entrada do nó hóspede.



(a) Nós sucessores e predecessores.

(b) Encaminhamento de pacotes.

Figura 2.14. Funcionamento do ROFL intradomínio.

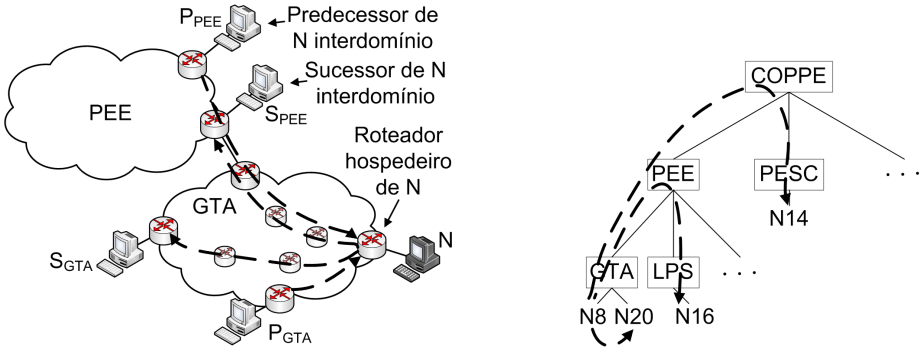
Os roteadores hospedeiros podem usar memória cache para armazenar ponteiros para identificadores de nós. Cada ponteiro é uma referência aos caminhos para os nós sucessores e predecessores dos seus nós hóspedes. Os roteadores hospedeiros também podem armazenar ponteiros referentes a caminhos que passam através dele vindos de outros nós da rede que não sejam nem sucessores ou predecessores dos seus hóspedes. O primeiro tipo de caminho possui precedência sobre o posterior, uma vez que a cache é limitada em tamanho. Assim como no Chord, o roteamento é realizado de maneira gulosa auxiliada por uma tabela de roteamento que possui ponteiros para identificadores de nós. Logo, um pacote enviado para um determinado identificador é encaminhado através do ponteiro mais próximo daquele identificador de destino. No pior caso, um pacote pode ser encaminhado ao longo de todos os sucessores. O conceito de proximidade entre ponteiros está relacionado com a proximidade entre identificadores. Os roteadores hospedeiros mantêm uma lista ordenada de identificadores e sempre que vão encaminhar

um pacote, eles comparam o identificador com a sua lista. O roteador que possuir o identificador menor mais próximo do identificador do destino é utilizado como próximo salto. A Figura 2.14(b) ilustra o encaminhamento de pacotes salto-a-salto. O pacote é encaminhado entre a origem N e o destino D passando por todos os nós sucessores dos roteadores hospedeiros no caminho.

O ROFL define procedimentos para casos de falhas de roteadores. Em caso de falha, os roteadores vizinhos inspecionam todos os seus ponteiros em cache e enviam mensagens de falha ao longo de qualquer caminho que possua o roteador com problemas. Se for um problema de um nó hóspede, o roteador envia mensagens de falha aos predecessores e sucessores do nó que falhou. Quando uma mensagem de falha chega a roteadores hospedeiros, eles recuperam os nós sucessores e predecessores do nó com problemas na tentativa de manter a rede conectada. Entretanto, nem sempre isso é possível. Por isso, os roteadores sempre distribuem rotas aos roteadores mais estáveis para evitar particionamentos da rede. Os roteadores localmente desempenham checagens de correção baseados no conteúdo das mensagens recebidas e então executam protocolos de recuperação de particionamentos para assegurar que a rede convirja em um único espaço de endereçamento.

No interdomínio, a operação do ROFL é semelhante à do intradomínio, exceto por considerar políticas no nível de ASes. Tais políticas podem ser modeladas como um grafo de ASes organizado de maneira hierárquica. Cada AS forma um espaço de identificação circular e os diferentes ASes se comunicam através da estrutura em árvore interdomínio. Para possibilitar a comunicação entre os diferentes ASes, os espaços de identificação devem se unir. O processo de união entre ASes diferentes pode ser dividido em três etapas. Na primeira etapa, cada AS deve descobrir todos os outros ASes hierarquicamente superiores com os quais possui acordos. Na segunda etapa, os ASes se unem recursivamente, assim como no Canon, utilizando os roteadores que compartilham enlaces em comum em ASes diferentes. Na última etapa, cada AS define ponteiros para roteadores em ASes vizinhos que diminuam o caminho entre eles. Um determinado nó em um AS pode ser globalmente alcançável se o seu roteador hospedeiro mantiver predecessores e sucessores referentes em cada nível de hierarquia de ASes, como ilustrado na Figura 2.15(a). Semelhante ao caso intradomínio, no nível dos ASes o roteamento também é realizado pela fonte. Assim, um determinado pacote é encaminhado de acordo com o caminho de ASes pré-estabelecido armazenado nos roteadores hospedeiros como um vetor de caminho do BGP. Caso haja falha nos enlaces entre ASes, os ASes restabelecem a sub-árvore formada entre eles para garantir que a rede mantenha conectividade. Se isso não for possível, enlaces operacionais podem ser adicionados. As políticas entre diferentes sistemas autônomos permitem que alguns caminhos sejam tratados como caminhos de reserva.

O encaminhamento de pacotes no ROFL segue propriedades de isolamento. Uma comunicação entre nós no mesmo AS não utiliza ponteiros para nós externos àquele AS. Semelhantemente, uma comunicação entre nós em ASes diferentes resulta em pacotes encaminhados através do menor caminho possível na árvore. Logo, o caminho formado entre dois ASes deve seguir através do primeiro ascendente comum. A Figura 2.15(b) ilustra o isolamento entre os ASes no ROFL. Caso `N8` queira se comunicar com `N20`, ele o fará sem utilizar ponteiros externos. Entretanto, caso `N8` queira se comunicar com `N16`, `N8` precisa utilizar o nó sucessor e predecessor no domínio do PEE, subindo até o primeiro nível hierárquico em comum. Equivalentemente, a comunicação entre `N8` e `N14`



(a) Nós sucessores no ROFL.

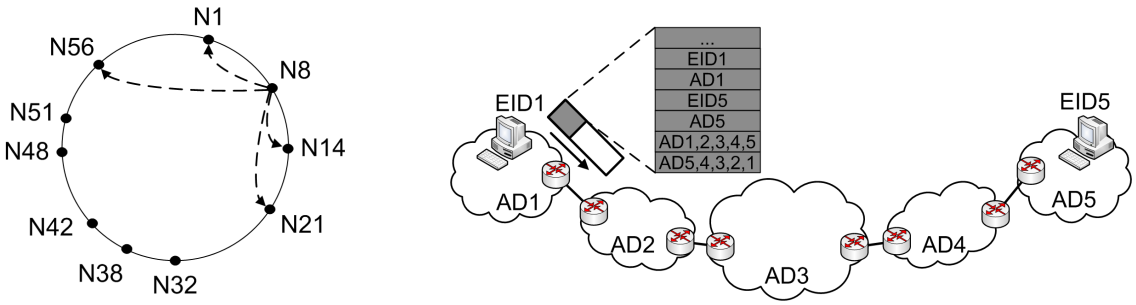
(b) Hierarquia de encaminhamento no ROFL.

Figura 2.15. Funcionamento do ROFL interdomínio.

utiliza o domínio da COPPE para a comunicação. Como visto, o encaminhamento de pacotes segue uma estrutura hierárquica. Isso demonstra que apesar de utilizar um espaço de identificação plano, questões como escalabilidade e divisão da Internet em ASes ainda requerem que o roteamento seja feito de maneira estruturada hierarquicamente.

Os problemas de escalabilidade do ROFL estão ligados à quantidade de ponteiros armazenados para outros nós da rede, à carga de controle para o estabelecimento dos espaços de identificação circulares, à latência da entrada dos nós e recuperação de falhas. Além do ROFL, outro exemplo de protocolo que planifica o espaço de identificação é o VRR (*Virtual Ring Routing*) [Caesar et al. 2006a]. O VRR, porém, limita o escopo do roteamento a uma rede sem-fio de múltiplos saltos. Assim, o VRR reduz o problema da escalabilidade já que lida com um número menor de nós se comparado à Internet. O funcionamento do VRR é semelhante ao do ROFL intradomínio. No VRR, entretanto, cada nó N mantém uma tabela de caminhos para os m vizinhos consecutivos mais próximos no espaço de identificação circular. Desses m vizinhos, metade possui identificadores no sentido horário e a outra metade no sentido anti-horário ao de N. A Figura 2.16(a) mostra os m vizinhos de `N8`, $m = 4$. Cada caminho é definido pelos identificadores de origem e destino e pelos identificadores do próximo salto na rede física. A tabela de caminhos do VRR é diferente da tabela de roteamento do ROFL pois armazena nós sucessores e predecessores consecutivos. Além disso, o VRR utiliza informações da rede física já que foi desenvolvido para uso em redes sem-fio, diferente do ROFL. No VRR, essa informação evita que vizinhos físicos cujos enlaces não ofereçam qualidade mínima sejam usados como próximos saltos.

Diferente do ROFL e do VRR, mas ainda na direção da planificação do espaço de endereçamento, está o AIP (*Accountable Internet Protocol*) [Andersen et al. 2008]. O AIP é outra proposta para roteamento na Internet que evita o uso de prefixos e endereços sem classes definidas (*Classless Inter-Domain Routing* - CIDR). Para isso, ele retorna à estrutura original de endereços da Internet que era a concatenação do identificador de rede com o da estação. Essa estrutura possuía dois níveis hierárquicos, um de rede, utilizado pelos roteadores para encaminhar pacotes, e outro de estações. Logo, O AIP utiliza uma abordagem com dois níveis hierárquicos, diferente do ROFL e do VRR. No AIP, o nível dos roteadores é chamado de domínio de responsabilização (*Accountability Domain* - AD) e o das estações é chamado de identificador de pontos finais (*End-point IDentifier*



(a) Vizinhos no espaço de identificação circular no VRR.

(b) Encaminhamento no AIP.

Figura 2.16. Outros protocolos de roteamento plano.

- EID). Essa divisão reduz os problemas de escalabilidade comparado ao ROFL, mas não desassocia completamente o identificador do nó do identificador da rede. Um dos problemas enfrentados pelo AIP é como relacionar os ASes aos domínios de responsabilização. Na Internet, os ASes possuem identificadores totalmente desassociados dos prefixos de endereços anunciados por eles. No AIP, o identificador do domínio de responsabilização é utilizado nos dois casos.

A estrutura de endereçamento do AIP é formada pela concatenação do identificador do domínio (AD) com o identificador do nó (EID). O AD e o EID são formados a partir de saídas de uma função *hash*. Ambos são calculados sobre a chave pública que identificam o domínio e a estação, respectivamente. Essa característica permite que os endereços possam ser autenticados. As conexões TCP são estabelecidas utilizando apenas o identificador do nó, assim há a possibilidade dos nós se deslocarem entre domínios diferentes. O único requisito é a garantia da unicidade do identificador do nó, conquistada utilizando parte do endereço MAC no identificador. O encaminhamento de pacotes no AIP é feito utilizando roteamento pela fonte no nível dos domínios. Isso ocorre porque tanto a ausência de organização hierárquica dos domínios quanto o não emprego de prefixos de rede impedem que o encaminhamento seja feito por técnicas de busca de melhor prefixo (*best match prefix*). A Figura 2.16(b) mostra o caminho percorrido por um pacote desde sua origem EID1 até o seu destino EID5. O cabeçalho do pacote possui entre outros campos, os identificadores do nó e do domínio de origem, os identificadores do nó e do domínio do destino e as pilhas de identificadores de domínios entre a origem e o destino tanto no caminho direto quanto no reverso.

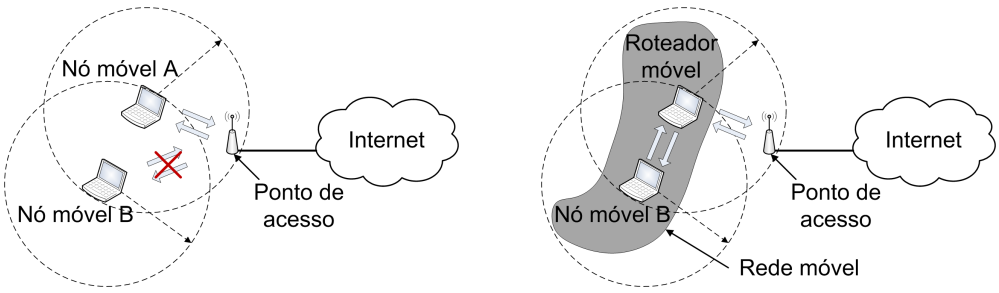
O AIP, assim como o ROFL, ainda possui problemas de escalabilidade e aplicabilidade na Internet. Em ambos os casos, a infraestrutura da Internet desde sistemas finais até roteadores precisariam ser alterados para adotar qualquer uma das propostas. Portanto, o emprego dessa estratégia ainda não é óbvio apesar de solucionar problemas como a mobilidade de estações.

2.4.3. Mobilidade de Rede

A mobilidade está normalmente associada à mobilidade de uma única estação e não à mobilidade de uma rede, onde uma rede é um conjunto de uma ou mais estações conectadas. Um exemplo disso é o modelo base do IP móvel que define a existência de um agente estrangeiro, um agente domiciliar, uma estação correspondente e uma estação

móvel. Esse modelo representa uma limitação já que a popularização das redes sem-fio requer que o suporte à mobilidade seja estendido aos casos nos quais redes inteiras se movem e não apenas uma única estação. Exemplos típicos de redes móveis são as redes ad hoc, as redes veiculares [Alves et al. 2009] e até mesmo as redes formadas entre dispositivos de comunicação baseados no IP carregados por pessoas (*Personal Area Network* - PAN). Em todos esses cenários as diferentes estações podem se deslocar em conjunto e podem estar conectadas entre si com acesso à Internet. É válido ressaltar que o mais importante é manter as conexões previamente estabelecidas ativas mesmo em face da troca do ponto de interconexão à Internet.

No modelo base do IP móvel, cada estação da rede é tratada como uma estação isolada. Caso uma das estações não tenha acesso direto ao ponto de interconexão, ou ponto de acesso, essa estação perde a conectividade com a rede, como visto na Figura 2.17(a). Propostas como o NEMO (*NEtwork MObility*) [McCarthy et al. 2009] e as suas variantes NEMO+ [McCarthy et al. 2008b], Light-NEMO+ [Sabeur et al. 2006] e MANEMO (*MAnet* NEMO) [McCarthy et al. 2009, McCarthy et al. 2008a] vêm sendo realizadas para possibilitar que mesmo os nós sem acesso direto ao ponto de interconexão permaneçam conectados e com acesso à Internet. Assim, a conectividade pode ser garantida não somente para uma estação, mas para uma rede inteira. De maneira geral, o NEMO define que uma das estações da rede móvel deve ser escolhida para funcionar como um roteador que provê acesso à Internet a todas as outras estações da rede móvel. Assim, as estações que não possuem acesso direto ao ponto de interconexão devem encaminhar os seus pacotes até esse roteador, denominado roteador móvel (*Mobile Router* - MR). A Figura 2.17(b) ilustra simplificadamente o funcionamento das propostas para mobilidade de rede, na qual o nó móvel A da Figura 2.17(a) exerce a função de roteador móvel.



(a) Modelo básico do IP móvel. (b) Mobilidade de rede.

Figura 2.17. Conectividade em redes sem-fio móveis.

O IETF (*Internet Engineering Task Force*) criou o grupo de trabalho NEMO para trabalhar no tema de mobilidade de rede [Perera et al. 2004]. O grupo de trabalho NEMO é responsável pelo desenvolvimento e padronização dos protocolos para suporte da mobilidade de rede sobre o IP. O projeto inicial é fortemente baseado no IP móvel. Dessa forma, o NEMO pode ser implementado como uma extensão do IP móvel, na qual as funcionalidades definidas pelo IP móvel para uma estação são transferidas para o roteador móvel. Como consequência, as alterações após eventuais mudanças de ponto de interconexão com a Internet são tratadas apenas pelo roteador móvel. Tais alterações, como mudança de endereço IP (*Care-of-Address* - CoA) e restabelecimento do túnel com

o agente domiciliar, tornam-se transparentes aos outros nós da rede móvel. Com exceção do roteador móvel, todos os outros nós da rede mantêm suas configurações constantes. O NEMO também utiliza agentes domiciliares como o IP móvel para manutenção da conectividade. No NEMO, os roteadores móveis também enviam mensagens de atualização aos seus agentes domiciliares. Entretanto, os agentes domiciliares associam o endereço do roteador móvel na rede estrangeira (CoA) a um prefixo de rede ao invés de apenas a um endereço IP. Assim, todos os pacotes recebidos pelo agente domiciliar destinados a um dos nós da rede móvel são encapsulados e encaminhados através do túnel até o roteador móvel. O roteador móvel, por sua vez, desencapsula os pacotes recebidos e os encaminha até os seus respectivos destinos na rede móvel. Além disso, o roteador móvel também encaminha para a Internet todos os pacotes oriundos da rede móvel. Caso haja filtragem de egresso na rede estrangeira, os pacotes são encapsulados e encaminhados através do túnel no sentido reverso até o agente domiciliar. O agente domiciliar, por fim, desencapsula os pacotes e os encaminha via roteamento IP até os nós correspondentes na Internet. A Figura 2.18 ilustra o caminho seguido pelos pacotes usando o NEMO.

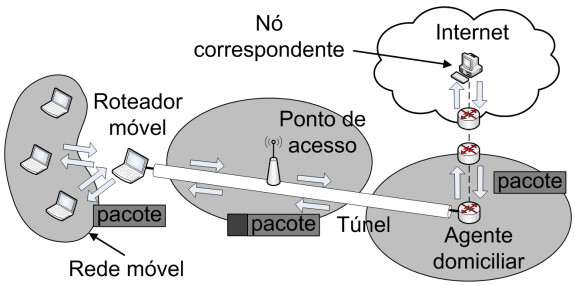


Figura 2.18. Encaminhamento de pacotes no NEMO.

O grupo de trabalho responsável pelo NEMO padronizou inicialmente um protocolo de suporte básico (NEMO *Basic Support Protocol* - NEMO BS) para o IPv6 e tem abordado diferentes problemas em outros documentos, inclusive a implementação do NEMO sobre o IPv4 [Devarapalli et al. 2005]. A sinalização utilizada pelo NEMO BS é uma extensão das mensagens definidas pelo IP móvel. As mensagens do NEMO BS possuem uma *flag* adicional que identifica se a mensagem foi enviada por um roteador móvel ou por um nó móvel. Essas mensagens são enviadas utilizando o cabeçalho de extensão de mobilidade, no caso do IPv6; ou mensagens de controle sobre o UDP, no caso do IPv4. As mensagens mais utilizadas pelo NEMO BS são as mensagens relativas à atualização de localização do roteador móvel (*binding updates*). As mensagens de atualização de associação com as redes estrangeiras e os respectivos reconhecimentos são utilizadas para notificar os agentes domiciliares do novo ponto de interconexão com a Internet (CoA). As mensagens de atualização contêm o novo CoA, a *flag* adicionada pelo NEMO BS e o prefixo da rede móvel. Este último, porém, é uma opção que depende do modo de operação do NEMO BS, que pode ser implícito ou explícito. No modo implícito, as mensagens de atualização não contêm o prefixo da rede móvel. Nesse caso, os agentes domiciliares descobrem o prefixo de alguma maneira não definida pelo protocolo de suporte básico. Já no modo explícito, o prefixo da rede é adicionado às mensagens de atualização. Uma vez recebida a mensagem de atualização, independente do modo, os agentes domiciliares enviam o reconhecimento positivo correspondente. A Figura 2.18

ilustra como o NEMO BS especifica o roteamento dos pacotes enviados e recebidos pelos nós da rede móvel. Os roteadores móveis possuem uma interface IPv6 ou IPv4 pertencente à rede móvel, interface de ingresso que pode ser configurada de maneira estática; e outra interface conectada à rede que oferece conexão de saída para a Internet, interface de egresso. A interface de egresso é registrada na rede domiciliar a partir da associação realizada com o endereço IP da rede estrangeira (CoA). Um túnel bidirecional é estabelecido entre o roteador móvel e o seu agente domiciliar. O agente domiciliar, ao invés de somente encaminhar o tráfego recebido destinado ao roteador móvel através do túnel, encaminha também todo tráfego do prefixo de rede associado ao roteador móvel.

O protocolo de suporte básico NEMO BS deve lidar com desafios relacionados com o ambiente de operação e com a forma de implementação escolhida para estender o IP móvel. Desafios relacionados com a segurança e com o desempenho devem ser considerados. Por exemplo, para garantir que a origem dos pacotes enviados através do túnel seja verdadeira, tanto o roteador móvel quanto o agente domiciliar devem checar se o endereço IP de origem dos pacotes é um endereço IP pertencente à faixa de endereços da rede domiciliar. Embora os desafios relacionados com a segurança sejam relevantes, muitos trabalhos na área visam aumentar o desempenho do NEMO. Um dos problemas frequentemente abordados é a ineficiência do roteamento. Como visto, caso um nó da rede móvel deseje se comunicar com um nó na Internet, chamado de nó correspondente, todos os seus pacotes devem ser enviados primeiramente ao seu agente domiciliar. O agente domiciliar, então, encaminha os pacotes até o nó da Internet, o que torna o roteamento sub-ótimo. O problema pode ser agravado considerando que um único roteador móvel pode ter associado a ele mais de uma rede móvel. Fundamentalmente, nada impede que coexistam mais de uma rede móvel e que o roteador móvel de uma rede utilize o roteador móvel de outra rede para se comunicar com a Internet. Nesse caso, o NEMO é conhecido como *Nested* NEMO [McCarthy et al. 2008b] e as comunicações podem ser bastante ineficientes visto que todo tráfego enviado pelos roteadores móveis devem sempre passar pelos seus agentes domiciliares. A Figura 2.19 ilustra o problema.

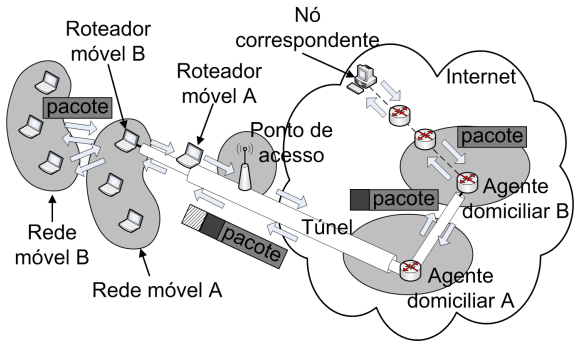
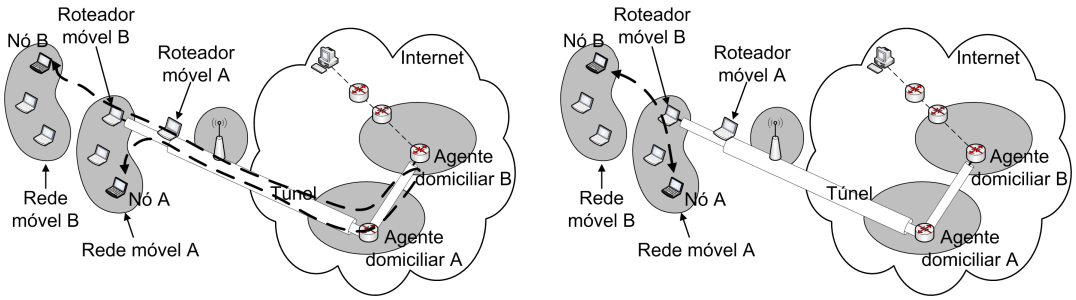


Figura 2.19. Problema conhecido como *Nested* NEMO.

Propostas para aumentar a eficiência do roteamento utilizam caminhos diretos entre os nós da rede móvel e os nós da Internet e até mesmo entre os nós em redes móveis diferentes. Algumas propostas analisam a possibilidade do uso de roteamento pela fonte para evitar que os pacotes passem sempre pelos agentes domiciliares. O NEMO+ [McCarthy et al. 2008b] propõem três protocolos para tornar as comunicações

envolvendo nós de redes móveis vizinhas mais eficientes. Esse tipo de comunicação ocorre quando uma rede móvel não possui acesso direto à Internet, e assim utiliza a rede vizinha para encaminhar o seu tráfego, e quando os nós móveis de diferentes redes desejam se comunicar. O primeiro protocolo proposto pelo NEMO+ é o TD (*Tree Discovery*). Esse protocolo é utilizado para auxiliar os roteadores móveis a escolherem qual dos roteadores móveis vizinhos oferece o melhor caminho até a Internet. A escolha é baseada nos anúncios IPv6 de descoberta de vizinhança enviados pelos vizinhos. Cada anúncio contém o caminho utilizado até o ponto de interconexão com a Internet obtido através de mensagens ICMPv6. Esse caminho representa um ramo da árvore formada desde o ponto de interconexão da rede (raiz) e os roteadores móveis vizinhos (folhas). Ao receber o anúncio, cada roteador escolhe o ramo da árvore que mais lhe convém se associar. Além disso, as mensagens de descoberta de vizinhança podem ainda ser utilizadas para evitar a formação de laços de roteamento (*loops*). Outro protocolo, o NINA (*Network In Node Advertisement*) é utilizado para anunciar aos roteadores móveis localizados em posições mais próximas ao ponto de interconexão os prefixos das sub-redes associados a cada roteador móvel. Esses anúncios possibilitam que as comunicações entre nós móveis associados a diferentes redes da mesma árvore possam se comunicar sem a necessidade do uso dos agentes domiciliares. A Figura 2.20 ilustra o caminho percorrido pelos pacotes do nó B na rede móvel B até o nó A na rede A com e sem o uso do protocolo NINA.



(a) Encaminhamento sem o protocolo NINA. (b) Encaminhamento com o protocolo NINA.

Figura 2.20. Encaminhamento com e sem o uso do protocolo NINA.

O último protocolo, chamado RRH (*Reverse Routing Header*), é usado para tornar o roteamento de pacotes enviados para a Internet também mais eficiente. No RRH, todo roteador móvel antes de encaminhar um pacote para a Internet atualiza o endereço IP de origem do pacote. O procedimento de atualização estabelece que o endereço IP de origem deve ser o do roteador móvel que encaminha o pacote e o endereço de origem anterior deve ser armazenado em uma lista no cabeçalho. Essa lista contém o endereço IP de todos os roteadores móveis já atravessados. Como o endereço IP de origem do pacote que é recebido pelo ponto de interconexão é do último roteador móvel, o pacote será somente encaminhado até o agente domiciliar desse último roteador antes de ser enviado até o nó correspondente. O RRH define que o cabeçalho contendo a lista de endereços IP de todos os roteadores móveis deve ser adicionado pelo agente domiciliar no caminho reverso para que o pacote possa ser entregue ao nó móvel de origem. Caso contrário, o pacote chegaria ao último roteador móvel e não seria possível identificar a verdadeira origem do pacote. O Light-NEMO+ [Sabeur et al. 2006] possui uma abordagem um pouco

diferente do protocolo RRH. No Light-NEMO+, o último roteador armazena um identificador do fluxo, semelhante a um *cookie*, e encaminha o pacote até o nó correspondente na Internet. Os pacotes cuja fonte ou destino estejam na rede móvel também carregam o *cookie* de identificação do fluxo. Assim, todo pacote vindo da Internet recebido pelo último roteador móvel possui uma identificação que pode usada para mapeamento do pacote para o fluxo correspondente. Note que o protocolo RRH usava roteamento pela fonte para identificar o destino ou a origem do pacote na rede móvel. Já o Light-NEMO+ utiliza um *cookie* e, portanto, dispensa o uso do roteamento pela fonte.

O MANEMO (*MAnet* NEMO) [McCarthy et al. 2008a] é um protocolo que tem por objetivo garantir que os nós móveis de uma rede ad hoc possam sempre ser alcançados de qualquer ponto na Internet. O roteador móvel que executa o protocolo MANEMO possui sua interface de ingresso configurada conforme a rede estrangeira visitada e a interface de egresso configurada conforme a rede ad hoc a qual faz parte. A interface de egresso executa um protocolo de roteamento ad hoc, p. ex. o OLSR [Clausen et al. 2001] configurado para se anunciar como *gateway* para a Internet. A arquitetura unificada MANEMO (*Unified MANEMO Architecture* - UMA) [McCarthy et al. 2009] é uma proposta para unificar os protocolos NEMO e assim garantir conectividade permanente aos nós da rede ad hoc. A UMA define a maneira pela qual os diferentes nós móveis conectam a Internet, via acesso direto a pontos de interconexão ou via outras redes móveis, e como os túneis entre os diferentes agentes domiciliares são estabelecidos. É responsabilidade dos agentes domiciliares identificarem os túneis criados para comunicação com cada um dos nós móveis assim como o restabelecimento da comunicação caso haja mudanças no posicionamento desses nós. A alteração do posicionamento dos nós móveis pode levar a mudanças do roteador móvel e, consequentemente, dos túneis previamente estabelecidos. A arquitetura UMA deve lidar com a dinamicidade da rede.

2.4.4. Múltiplos caminhos

A Internet atual é baseada em algoritmos de roteamento de caminho único. Dessa forma, os protocolos intradomínio e, principalmente, os interdomínio representados pelo BGP anunciam aos seus vizinhos apenas uma única opção de caminho para cada destino da rede. Uma maneira de contornar essa limitação e implementar os múltiplos caminhos é através de roteamento pela fonte. A definição de rotas a priori é uma solução para utilizar caminhos diferentes do padrão anunciado pelos protocolos de roteamento. Um exemplo de tecnologia que usa roteamento pela fonte é o PNNI (*Private Network-to-Network Interface* ou *Private Network Node Interface*) utilizado no ATM. O PNNI divide a topologia da rede em diferentes níveis hierárquicos e define por quais roteadores de cada nível os pacotes devem ser encaminhados [Kaur et al. 2003].

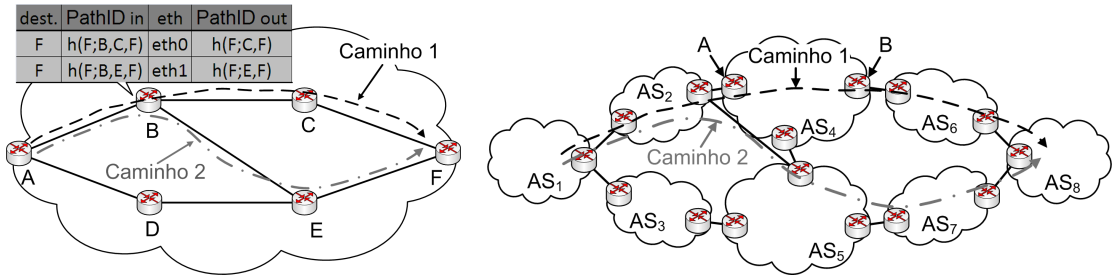
O arcabouço BANANAS [Kaur et al. 2003] é outra proposta que utiliza roteamento pela fonte. O BANANAS usa o conceito de `PathIDs` para identificação de caminhos. O `PathID` é a saída de uma função *hash* dos identificadores dos vértices e dos enlaces que compõem o caminho entre dois nós quaisquer em uma rede. Os identificadores dos vértices, p. ex. endereços IP, dos enlaces e de ASes são globalmente conhecidos e, portanto, o `PathID` também. O conceito de "globalmente conhecido" varia se o escopo for intradomínio (identificadores de enlaces e vértices) ou interdomínio (identificadores de ASes). No caso intradomínio, o conhecimento global é conquistado por algoritmos

de roteamento baseados em estado do enlace, já no caso interdomínio, pelos vetores de caminho do BGP. Para evitar colisões entre `PathIDs`, o `PathID` utilizado é estendido a uma tupla formada pelo endereço IP do destino e o resultado da *hash* calculado. Os `PathIDs` são adicionados a cada pacote para serem usados posteriormente durante o encaminhamento.

O BANANAS considera o roteamento intra e interdomínio. Vale mencionar que o BANANAS pode operar mesmo se apenas uma parte dos roteadores implementar o arcabouço proposto. Para isso, o cálculo do roteamento é realizado com restrições, pois considera como possíveis caminhos aqueles que passam por roteadores que implementam o BANANAS. No intradomínio, o roteador fonte envia os pacotes pelos múltiplos caminhos encontrados, como ilustrado na Figura 2.21(a). Essa figura mostra a tabela de encaminhamento do roteador B e o encaminhamento realizado para pacotes de mesma origem e destino, mas que seguem caminhos diferentes na rede. A função *hash* usada para o cálculo dos `PathIDs` é denotada por $h(.)$ na figura. Os pacotes carregam o `PathID` correspondente ao caminho seguido. Um roteador intermediário que implementa o BANANAS utiliza para o encaminhamento, ao invés da tupla prefixo do endereço de destino, próximo salto e interface de saída; a tupla prefixo do endereço de destino, `PathID` de entrada, interface de saída e `PathID` de saída. O `PathID` de entrada é a *hash* de todos os identificadores dos roteadores a partir do roteador atual até o destino e o `PathID` de saída é a saída da função *hash* dos identificadores desde o próximo salto até o destino. Ao receber um pacote, um roteador busca a entrada correspondente em sua tabela de roteamento baseado no prefixo do endereço de destino e no `PathID` de entrada. Antes de encaminhar o pacote, o roteador substitui o `PathID` do pacote pelo `PathID` de saída. Assim, o próximo roteador no caminho pode repetir o mesmo procedimento de encaminhamento. No BANANAS, todos os roteadores devem executar um algoritmo que calcule múltiplos caminhos já que o arcabouço foi proposto com essa finalidade. Mesmo os roteadores intermediários devem conhecer todos os possíveis caminhos a partir dele próprio até os possíveis destinos da rede para encaminhar pacotes seguindo a rota escolhida pelas fontes. O efeito do armazenamento dos múltiplos caminhos é o aumento das tabelas de roteamento. Entretanto, o BANANAS aborda esse compromisso utilizando técnicas de codificação para armazenamento de informações compactadas.

O roteamento interdomínio é uma abstração do roteamento intradomínio substituindo roteadores por ASes. O funcionamento é bastante semelhante se considerado o papel dos roteadores de borda de entrada e saída do AS semelhante ao papel das interfaces de entrada e saída de um roteador. Um `PathID` é adicionado aos pacotes, nesse caso chamado de `e-PathID`, no qual os identificadores são os identificadores dos sistemas autônomos do caminho. Um roteador de borda de entrada de um AS ao receber um pacote examina o prefixo do endereço de destino e encaminha até o roteador de borda de saída correspondente. Para isso, o endereço IP do roteador de borda de saída é utilizado como endereço de destino do pacote. O endereço de destino original é armazenado em uma estrutura em pilha na qual o endereço do roteador de borda de saída é inserido no topo. O roteador de borda de saída retira o seu endereço do topo da pilha e reinsere o endereço IP original do destino no pacote. O roteador de borda de entrada do próximo AS repete o procedimento de empilhamento de endereços IP. O `PathID` utilizado pelo roteamento em cada AS deve ser calculado baseado no caminho intradomínio escolhido. O uso de

múltiplos caminhos interdomínio deve lidar com questões relacionadas a políticas e acordos entre ASes. Essas questões não afetam o encaminhamento intradomínio que pode ser realizado sempre através de múltiplos caminhos quando disponíveis. A Figura 2.21(b) ilustra um exemplo de múltiplos caminhos interdomínio entre fontes no AS_1 e destinos no AS_8. Ainda na figura, os roteadores A e B representam o roteador de borda de entrada e saída, respectivamente, do AS_4 com relação ao fluxo do caminho 1. Logo, um pacote seguindo o caminho 1 utiliza como endereço IP de destino o endereço do roteador B após ser encaminhado por A.



(a) Roteamento intradomínio. (b) Roteamento interdomínio.

Figura 2.21. Encaminhamento de pacotes conforme o arcabouço BANANAS.

Redes sobrepostas também podem ser utilizadas para prover múltiplos caminhos. A vantagem dessa abordagem é não exigir modificações dos equipamentos do núcleo da rede. Os múltiplos caminhos são escolhidos na camada sobreposta e o encaminhamento dos pacotes segue o processo tradicional da Internet. A arquitetura RON (*Resilient Overlay Network*) [Andersen et al. 2001], inicialmente proposta para recuperação de falhas de rede, pode ser utilizada também para proporcionar múltiplos caminhos na Internet. Entretanto, uma das principais desvantagens do seu uso, assim como do roteamento pela fonte, é a redução do controle das rotas escolhidas por parte dos ASes de trânsito. É interessante para esses ASes escolherem os vizinhos para engenharia de tráfego e estabelecimento de acordos comerciais. O protocolo MIRO (*Multipath Interdomain ROuting*) [Xu e Rexford 2006] oferece maior flexibilidade aos ASes de trânsito e, por isso, não usa nem redes sobrepostas nem roteamento pela fonte. O MIRO baseia-se na negociação entre ASes vizinhos para uso de múltiplos caminhos. Embora possivelmente os múltiplos caminhos existam, cada AS anuncia apenas o caminho que mais lhe convém por questões de políticas, implementação dos protocolos e escalabilidade. Entretanto, o MIRO argumenta que um determinado AS deve requisitar os múltiplos caminhos, caso tenha interesse. Assim, problemas de escalabilidade são contidos e a implementação em toda Internet pode ser feita de maneira gradual. Um AS que não use o MIRO não responde às requisições por caminhos alternativos. O encaminhamento, nesse caso, acontece como na Internet atual. A Figura 2.22 ilustra o procedimento de negociação entre os roteadores A e B por caminhos alternativos. O roteador A pertencente ao AS_2 não deseja encaminhar o seu tráfego através do caminho padrão anunciado pelo AS_4 que é através do AS_6. O AS_4 oferece um caminho alternativo através do AS_5 que é aceito pelo AS_2. Os ASes podem fazer requisições para mais de um AS vizinho para obter mais caminhos alternativos e os ASes podem anunciar caminhos alternativos conhecidos caso atenda uma requisição recebida. Assim, um AS pode utilizar os caminhos alternativos dos seus vizinhos.

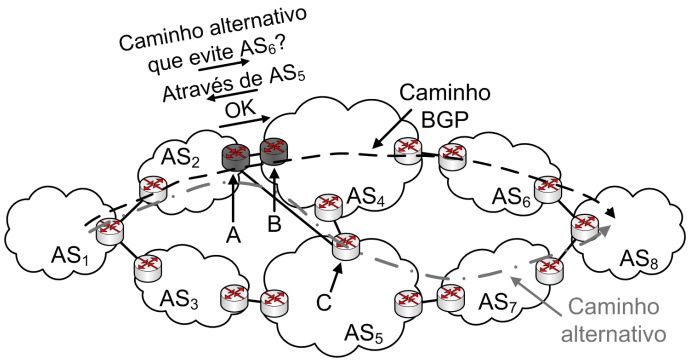


Figura 2.22. Negociação do MIRO para conhecimento de caminhos alternativos.

No MIRO, os caminhos descobertos pelo BGP padrão continuam a ser utilizados e os alternativos são usados através de procedimentos de tunelamento [Xu e Rexford 2006]. Logo, os pacotes que são enviados via caminho alternativo são enviados através de um túnel formado entre roteadores dos ASes vizinhos para garantir que o caminho alternativo seja, de fato, utilizado. O identificador do túnel é enviado pelo AS mais próximo ao destino para o AS mais próximo à fonte após o término do procedimento de negociação. Na Figura 2.22, um túnel entre os roteadores A e C é formado para encaminhamento de pacotes pelo caminho alternativo. Os caminhos alternativos podem ser utilizados para divisão de tráfego, no qual o tráfego mais prioritário utiliza o caminho alternativo, e para balanceamento de carga. A maneira como o tráfego é dividido depende das políticas adotadas pelos ASes e pelos acordos entre eles.

O processo de tunelamento pode ser chamado também de encaminhamento através de pontos de deflexão [Wetherall 2006, He e Rexford 2008]. O ponto de deflexão é o roteador para o qual o caminho é desviado. No exemplo da Figura 2.22 o ponto de deflexão é o roteador C visto que o caminho BGP convencional levaria o tráfego através do roteador B no AS4. Após o ponto de deflexão, o tráfego pode seguir o caminho BGP convencional caso nenhum caminho alternativo adicional seja usado. O emprego de pontos de deflexão pode ocorrer também nas redes sobrepostas. Em casos extremos, o ponto de deflexão poderia ser a estação de um usuário em outro AS que estivesse participando do roteamento. A participação de usuários pode impactar na escalabilidade da rede, mas evita que modificações sejam feitas na rede. O ponto de deflexão pode ser escolhido pela fonte do tráfego. Entretanto, independente da situação, o destino do túnel deve estar ciente do tunelamento para encaminhar o tráfego ao destino correto. Outra maneira de usar caminhos alternativos sem o uso de tunelamento é através da inserção de rótulos (*tags*) nos pacotes [Motiwala et al. 2008]. Um usuário final que queira que os seus pacotes sejam encaminhados através de caminhos alternativos deve inserir um rótulo no pacote referente ao caminho desejado. Os rótulos podem variar de acordo com as propriedades do caminho alternativo. Caso um roteador não reconheça o rótulo ou não implemente o sistema, ele pode encaminhar o pacote pelo caminho padrão.

2.4.5. Escalabilidade na Internet

Muitas propostas para interconexão de redes na Internet do futuro possuem impacto direto no número de entradas nas tabelas de roteamento. Por exemplo, o uso de

múltiplos caminhos requer mais de uma entrada nas tabelas para um mesmo destino. Já o uso dos múltiplos domicílios requer que uma rede seja identificada por mais de uma faixa de endereços possivelmente disjuntos prejudicando a agregação. O uso de identificadores planos, como os usados nas técnicas de *Loc/ID Split* e roteamento plano, também dificulta a agregação de rotas já que o espaço de endereçamento não é organizado hierarquicamente. Os identificadores planos, em especial, ainda aumentam o espaço de endereçamento da Internet o que pode tornar o problema da escalabilidade ainda mais grave. O aumento do espaço de endereçamento também pode ser uma consequência do uso do IPv6 se os problemas de agregação persistirem. Além do número de entradas nas tabelas de roteamento, outro problema que afeta a escalabilidade na Internet é o número de mensagens de atualização de roteamento enviadas principalmente por ASes de borda. Essas mensagens são enviadas para todos os outros ASes e o número pode aumentar devido a configurações mal feitas ou a ações maliciosas [Massey et al. 2007, Jen et al. 2008].

O aumento acelerado do número de entradas nas tabelas de roteamento pode impactar significativamente a capacidade de armazenamento da maioria dos roteadores atuais. Tal impacto pode causar inconsistências entre as tabelas, ou até mesmo, problemas de funcionamento dos equipamentos de rede [Kim et al. 2009]. Já o problema do número excessivo de mensagens de controle pode causar sobrecarga de tráfego e instabilidades nas tabelas de roteamento. Uma das propostas mais simples investigadas para conter ambos os problemas é a redução da flexibilidade de faixas de endereços disjuntas para múltiplos domicílios. A redução da flexibilidade utiliza duas estratégias. A primeira é a eliminação da possibilidade das redes de acesso utilizarem faixas de endereços diferentes da faixa de endereço dos seus ISPs. A segunda é a eliminação da possibilidade das redes de acesso utilizarem faixas de endereços desagregadas das faixas recebidas dos seus ISPs [Jen et al. 2008]. Nessa proposta, as redes de acesso somente podem utilizar faixas de endereços pertencentes às faixas dos seus provedores. Ainda, as estações das redes de acesso que são multidomiciliadas devem receber múltiplos endereços IPs, onde os endereços pertencem à faixa dos ISPs diretamente conectados. Essa limitação permite que cada ISP anuncie prefixos de rede agregados. A Figura 2.23 ilustra esse tipo de abordagem. Entretanto, uma desvantagem dessa abordagem é que os administradores das redes de acesso devem concordar em utilizar apenas faixas de endereços dos seus provedores diretos, já que hoje já existe a possibilidade de usar faixas de endereços independentes do provedor. Como consequência, a redução das tabelas de roteamento na DFZ torna-se dependente dos interesses dos administradores. Essa dependência não existe na proposta de separação de endereços.

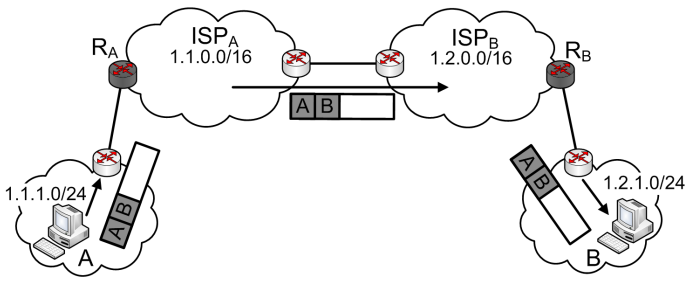


Figura 2.23. Encaminhamento em redes de acesso com sub-faixas de endereço de ISPs.

Uma proposta mais flexível para aumentar a escalabilidade na Internet é separar o espaço de endereçamento em Endereços Globalmente Roteáveis e Endereços Globalmente Entregáveis [Massey et al. 2007, Jen et al. 2008]. Os Endereços Globalmente Roteáveis são formados por endereços presentes nas tabelas de roteamento da DFZ e que são apenas alcançáveis na DFZ. Em oposição aos Endereços Globalmente Roteáveis, os Endereços Globalmente Entregáveis são os endereços de redes nas bordas da Internet que devem ser únicos e alcançáveis de qualquer lugar. Entretanto, os Endereços Globalmente Entregáveis não devem estar presentes nas tabelas da DFZ. A ausência de endereços de redes de borda nos roteadores da DFZ diminui o número de entradas e o número de prefixos anunciados pelo protocolo de roteamento interdomínio. Estimativas apontam que a eliminação dos prefixos de redes das bordas do roteamento interdomínio reduz o tamanho das tabelas e frequência de atualizações em até uma ordem de magnitude [Massey et al. 2007]. Além disso, um efeito indireto da separação de endereços é a possibilidade de emprego incremental de endereços diferentes do IPv4. Por exemplo, os endereços da borda podem ser IPv6 enquanto os do núcleo da rede podem ser IPv4.

A separação de endereços em dois tipos demonstra a divisão entre endereços de redes de ISPs, compostas tipicamente de roteadores de trânsito, e redes de acesso nas bordas. Assim, faixas de Endereços Globalmente Roteáveis são alocadas aos ISPs para que os diferentes ISPs sejam capazes de se comunicar. Um dado importante é que o número de ISPs tem sido estável em comparação ao número de redes de acesso [Massey et al. 2007]. Como consequência, as tabelas que contêm Endereços Globalmente Roteáveis não devem aumentar de tamanho rapidamente. Já os Endereços Globalmente Entregáveis são alocados para as redes de acesso. Esses endereços devem ser únicos para que cada rede de acesso ou estação seja identificada em toda Internet. Entretanto, como os provedores de serviço não conhecem os endereços das redes de acesso, este último não é globalmente roteável. Para que a correspondência seja possível, é necessário que haja um mapeamento dos dois tipos de endereços. Tal mapeamento é realizado nos roteadores de borda dos ISPs conectados às redes de acesso. Cada roteador de borda deve descobrir qual o mapeamento a realizar baseado no endereço de destino do pacote recebido. Após o mapeamento, o roteador encapsula o pacote e envia através do túnel formado até a roteador de borda da rede da estação do destino [Massey et al. 2007, Jen et al. 2008]. O roteador de borda da rede do destino desencapsula o pacote e entrega ao destino correspondente na rede de acesso.

O procedimento de mapeamento e encapsulamento [Jen et al. 2008] é ilustrado na Figura 2.24. Nessa figura, pacotes originados na estação A e destinados à estação B são enviados através de um túnel de R_A até R_B. A figura mostra também a divisão dos dois espaços de endereçamento existentes na proposta. É importante observar que os roteadores internos ao espaço de Endereços Globalmente Roteáveis não precisam conhecer os mecanismos de mapeamento e encapsulamento. Isso permite que a configuração deles permaneça a mesma que a atual, apenas com menos entradas nas tabelas de roteamento.

O uso do espaço de Endereços Globalmente Entregáveis permite que cada ISP agregue o maior número de faixas de endereços possível. Além disso, cada rede de acesso pode usar faixas de endereços independentes da faixa de endereços do ISP diretamente conectado, como visto na Figura 2.24. Essa característica facilita o uso dos múltiplos domicílios bem como a mudança das faixas de endereços utilizadas por cada rede de acesso. Entretanto, toda mudança realizada deve ser conhecida pelos roteadores responsáveis pelo

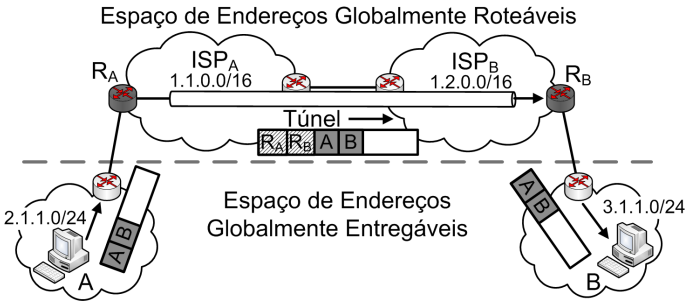


Figura 2.24. Encaminhamento utilizando dois espaços de endereçamento.

mapeamento para que esses continuem informando corretamente os outros roteadores na DFZ quais as redes de acesso que conhecem. As atualizações das informações de mapeamento podem acarretar problemas de escalabilidade uma vez que as atualizações são enviadas para em toda a DFZ.

Uma possibilidade para reduzir o número de mensagens de controle sobre o mapeamento é limitar as mensagens apenas para outros roteadores de borda utilizando técnicas, como por exemplo, o multicast. Outra possibilidade é o emprego de sistemas como o DNS para realizar o mapeamento. Todas as duas soluções reduzem a carga de controle, mas inserem outros problemas. Utilizar o multicast não é trivial na Internet e o uso de sistemas como o DNS pode inserir atrasos na resolução de endereços. Uma solução híbrida é proposta pela arquitetura APT (*A Practical Tunneling*) [Jen et al. 2009]. A arquitetura APT propõe que as informações de mapeamento sejam enviadas para todas as redes na DFZ. Entretanto, em cada uma das redes apenas um número reduzido de novos dispositivos de rede recebem tais informações. Esses dispositivos, denominados DM (*Default Mappers*), armazenam as tabelas completas com todos as informações sobre mapeamentos. Os roteadores de borda armazenam apenas as informações dos últimos mapeamentos realizados em uma memória cache. Logo, sempre que um pacote é recebido por um roteador de borda, ele verifica se o mapeamento específico está em sua memória cache. Caso esteja, o pacote é encapsulado e enviado, caso contrário, ele encaminha o pacote até o DM. O DM então trata o pacote como uma requisição para informação de mapeamento. Como resposta, o DM envia o mapeamento até o roteador de borda requisitante e, ao mesmo tempo, encapsula e encaminha o pacote em nome do roteador requisitante. Essa estratégia híbrida evita que os roteadores de borda armazenem tabelas completas, reduzindo o tamanho das tabelas utilizadas. Por outro lado, os DMs não fazem encaminhamento de muitos pacotes, reduzindo a necessidade de rapidez de encaminhamento. Essas características específicas permitem que os dispositivos sejam otimizados conforme a tarefa realizada.

O mapeamento deve lidar também com problemas de compatibilidade com equipamentos que ainda não separam o espaço de endereçamento [Vogt 2008]. Portanto, deve-se investigar maneiras de implementar a separação de endereços considerando a implementação gradual da proposta. Os roteadores de borda que não implementam a separação de endereços não realizam o mapeamento e o encaminhamento de pacotes. Além disso, eles podem não anunciar os prefixos das redes de acesso conhecidas. Logo, outros roteadores de borda descartam os pacotes recebidos cujo destino seja uma estação dessas redes de acesso que não tiveram seus prefixos anunciados. Isso ocorre porque os

roteadores de borda desconhecem o caminho até a rede do destino do pacote. Uma proposta para a adoção gradual da separação de endereços é o *Six/One Router* [Vogt 2008]. O *Six/One Router* é um protocolo para tradução de endereços entre estações em redes de acesso diferentes. A tradução é realizada no roteador de borda da rede de acesso de origem que mapeia o endereço de origem do pacote para um endereço de origem da DFZ. De maneira semelhante, o roteador de borda de origem realiza um procedimento de resolução de endereços para mapear o endereço de destino do pacote no endereço de destino na DFZ. Esse mapeamento é biunívoco, ou seja, cada endereço de uma estação em uma rede de acesso só pode ser mapeado em um endereço da DFZ. O endereço da DFZ pertence à faixa de endereços do ISP correspondente. O roteador executando o *Six/One Router* insere também uma extensão no cabeçalho do pacote para identificar os endereços de origem e destino das redes de acesso. Assim, quando o roteador de borda na rede de acesso do destino recebe o pacote, ele traduz os endereços para os endereços originais das redes de acesso e envia para a estação correspondente. O mapeamento é armazenado em memória cache para que o procedimento seja realizado mais rapidamente em pacotes seguintes. No caso de estações em redes de acesso legadas, ou seja, que não realizam a separação de endereços, o endereço da estação já é o próprio endereço da DFZ já que esses endereços têm que ser conhecidos globalmente. Portanto, caso um roteador de borda de origem realize a tradução, o endereço da DFZ da estação legada será o seu próprio endereço. No sentido reverso, caso uma estação legada seja a origem do pacote, o endereço de destino usado será o endereço da DFZ. Como o mapeamento é biunívoco, há somente uma estação associada a esse endereço da DFZ.

A Figura 2.25 ilustra o funcionamento do *Six/One Router* em caso de redes legadas. A estação A possui o seu endereço traduzido em um endereço do seu provedor de serviço. Entretanto, o endereço da estação B de destino é mantido, visto que a sua rede de acesso não realiza a separação de endereços. Logo, o endereço da estação B é conhecido globalmente e, por isso, pode ser mantido no pacote. A figura ilustra também a extensão incluída no cabeçalho do pacote pelo roteador R_A. Note que enquanto a rede de acesso da estação A não pode utilizar uma faixa de endereços do seu provedor (ISP_A), a estação B pode usar uma faixa do ISP_B.

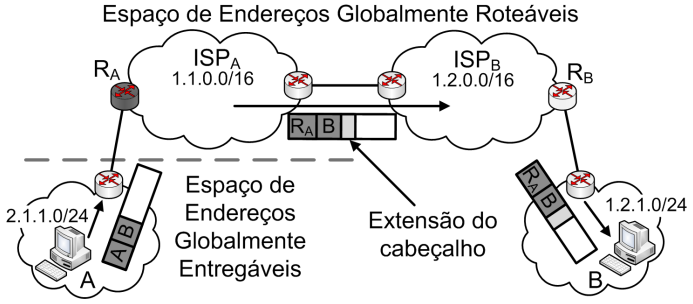


Figura 2.25. Funcionamento do *Six/One Router* em caso de redes de acesso legadas.

O HAIR (*Hierarchical Architecture for Internet Routing*) [Feldmann et al. 2009] é outra proposta para reduzir o número de entradas nas tabelas de roteamento da DFZ. Diferente das propostas de separação de endereços em dois níveis, o HAIR pode dividir os diferentes ASes em mais de dois níveis hierárquicos, onde o mais alto é composto

pelos principais ASes de trânsito da Internet e o mais baixo é composto pelas redes de acesso. Além disso, o HAIR utiliza técnicas para *Loc/ID Split* para possibilitar a mobilidade das estações. O encaminhamento de pacotes é realizado, portanto, baseado nos identificadores das estações de origem e destino que são mapeadas em endereços pela origem antes do envio dos pacotes. No HAIR, o endereço de origem codifica os identificadores de todos os roteadores de borda pelos quais o pacote deve passar desde sua origem até o nível hierárquico superior. Semelhantemente, o endereço de destino codifica os identificadores dos roteadores de borda desde o nível superior até o destino. O procedimento de encaminhamento é parecido com a opção *loose source routing* do IP. A diferença é que o caminho não é inserido no campo de opção do IP e é codificado nos endereços de origem e destino utilizados. O mapeamento do identificador do destino no endereço é realizado sob requisição. Para tal, assume-se que a estação de origem já conhece o identificador de destino e utiliza um sistema de resolução de identificação de estação em endereço. A Figura 2.26 mostra os endereços de origem e destino usados no pacote. Note que o endereço de origem é uma codificação do identificador da estação (estação A), do roteador do primeiro nível hierárquico acima (R_A) e do roteador do núcleo (R_{NA}). A divisão hierárquica dos ASes e o uso do roteamento pela fonte permitem que cada roteador conheça apenas as rotas até os roteadores de borda dentro do seu próprio nível hierárquico. Essa característica leva à redução do número de entradas nas tabelas de roteamento na DFZ.

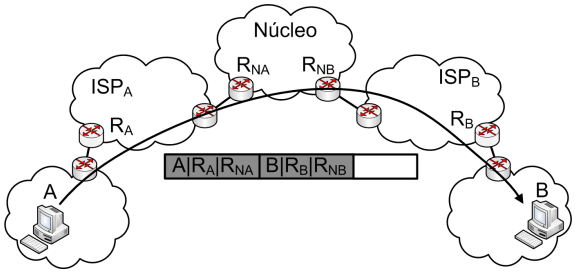


Figura 2.26. Funcionamento do HAIR.

Todos os trabalhos apresentados anteriormente reduzem o problema de escalabilidade na Internet através de novos esquemas de endereçamento. Entretanto, todas essas propostas necessitam alterar de certa forma a arquitetura atual de endereçamento, o que pode representar uma barreira para as suas implementações. Tendo em vista esse problema em curto e médio prazo, alguns trabalhos propõem a redução das tabelas de roteamento através de estratégias mais inteligentes de organização de prefixos ou de armazenamento em memória.

O ViAggre (*Virtual Aggregation*) [Ballani et al. 2008, Ballani et al. 2009] propõe distribuir a manutenção da tabela de roteamento intradomínio entre os roteadores de um ISP. Assim, cada roteador deve manter apenas parte da tabela completa de roteamento. O ViAggre divide o espaço de endereçamento em um conjunto de prefixos virtuais tal que cada prefixo virtual é maior que os prefixos agregados reais utilizados pelos roteadores. Por exemplo, todo o espaço de endereçamento conhecido por um ISP pode ser dividido em 128 prefixos virtuais /7 (0.0.0.0/7 até 254.0.0.0/7), no qual cada prefixo virtual corresponde a um conjunto de prefixos reais. Os prefixos virtuais não necessariamente possuem

correspondência com a topologia da rede, mas devem cobrir todos os possíveis prefixos reais para que não haja nenhuma rede inalcançável. As redes virtuais geradas a partir dos prefixos virtuais formam uma topologia que possui prefixos agregáveis e, portanto, tornam-se escaláveis. Para criar uma rede virtual, cada ISP define alguns roteadores para fazerem parte dessa rede. Esses roteadores mantêm rotas para todos os prefixos contidos no prefixo virtual. Tais roteadores são denominados pontos de agregação (*aggregation points*) para o prefixo virtual. Um ponto de agregação pode agregar mais de um prefixo virtual. Esse ponto de agregação deve apenas manter rotas para os prefixos contidos nos prefixos virtuais que ele mantém.

O encaminhamento de pacotes utilizando o ViAggre ocorre da seguinte maneira. Assim que o pacote entra na rede do ISP, esse pacote é encaminhado diretamente até o ponto de agregação mais próximo que mantém o prefixo virtual que engloba o prefixo do destino. Esse ponto de agregação possui uma rota para o prefixo do destino e, portanto, encaminha o pacote até o próximo ISP. O encaminhamento é realizado através de um túnel já que algum roteador no caminho pode desconhecer a rota escolhida e enviar o pacote de volta para o ponto de agregação. A divisão dos prefixos da Internet em prefixos virtuais é uma tentativa de rearrumar os prefixos para que eles se tornem agregáveis. Uma vez que os prefixos possam ser agregados, o número de entradas nas tabelas de roteamento diminui especialmente na DFZ.

Outras propostas atuam mais especificamente na organização da memória com relação aos tipos de entradas que devem ser armazenadas para reduzir o tamanho da tabela sem perder informação. Kim *et al.* [Kim et al. 2009] propõem armazenar em memória cache a tabela de roteamento apenas com as rotas usadas com maior frequência. As outras rotas são armazenadas em uma memória mais lenta e consultadas apenas em caso de ausência na cache. Caso um pacote seja recebido pelo roteador e uma rota não esteja disponível na memória cache, o roteador encaminha imediatamente o pacote em uma rota *default* e atualiza a sua memória cache baseado nas informações que possui em sua memória mais lenta. A atualização da cache segue a política de atualização adotada. O roteador que recebe o pacote enviado pela rota *default* conhece caminhos para todos os possíveis destinos. Esses roteadores podem ser projetados de maneira a armazenarem todas as possíveis rotas da rede. Além de armazenar parte da tabela de roteamento em cache, Kim *et al.* também propõem o uso de prefixos de mesmo comprimento. Através da análise de registros reais, os autores concluíram que prefixos /24 são os mais específicos possíveis que ainda não são filtrados pelos provedores por questões de segurança. Prefixos de comprimento menores, por exemplo /16, podem ser subdivididos em entradas com prefixos mais específicos que utilizam interfaces de saída diferentes da entrada com prefixo maior. Por exemplo, o prefixo 10.1.0.0/16 pode ter como interface de saída a eth0 enquanto o prefixo 10.1.1.0/24, a eth1. Caso um pacote destinado ao endereço 10.1.2.1 seja recebido, a entrada com prefixo maior é colocada em cache e o pacote é encaminhado pela eth0. Se em seguida um pacote com endereço de destino 10.1.1.1 for recebido, o pacote também é encaminhado pela eth0 ao invés da eth1 como definido pela entrada mais específica. A divisão em prefixos /24 é, portanto, uma tentativa de se evitar problemas durante o encaminhamento de pacotes ao utilizar parte das rotas em cache.

Organizar de maneira mais inteligente os prefixos nas tabelas de roteamento pode enfrentar dois desafios. O primeiro é que a solução proposta pode não ser definitiva.

Mudanças na arquitetura da Internet podem ser mais efetivas, porém mais difíceis de implementar sem incentivos consideráveis. Já o segundo problema é um compromisso verificado por Krioukov *et al.* [Krioukov et al. 2007] entre a redução das tabelas de roteamento e o aumento do tamanho médio das rotas. Krioukov *et al.* demonstram que, na Internet, o crescimento das tabelas de roteamento é logarítmico se forem utilizados algoritmos para compactação das tabelas. Entretanto, tal crescimento só é possível caso a Internet seja estática e os endereços estiverem relacionados com a topologia da rede. Se forem utilizados identificadores planos, o crescimento das tabelas de roteamento é polinomial, onde no melhor caso o crescimento é linear se os algoritmos de roteamento buscarem os caminhos mais curtos. Caso os caminhos encontrados possuam certa tolerância e não necessariamente sejam sempre os mais curtos, essa taxa de crescimento pode ser menor, no melhor caso.

2.4.6. Caminhos programáveis

A provisão da qualidade de serviço na Internet deve lidar com os diferentes requisitos das aplicações. Para possibilitar serviços diferentes do tradicional "melhor esforço", funcionalidades para visualização da topologia da rede bem como métodos sofisticados para projetos de novas aplicações são necessários. Essas novas técnicas podem ser utilizadas para mover funções das aplicações para a rede, já que o desempenho da rede pode variar [Clark et al. 2004]. Maneiras para monitorar o desempenho da rede, como movimentar as funções das aplicações e auxiliar o processo de escolha dos caminhos, podem ser feitas através da introdução de inteligência na rede. Tal inteligência é viabilizada através da aquisição de experiência e visão global. A inteligência é adquirida por agentes que atuam na rede e utilizam o conhecimento disponível, frequentemente obtido no Plano de Conhecimento [Clark et al. 2003], para tomar decisões sobre quais são os melhores caminhos conforme a aplicação. Além do emprego de agentes, a escolha do caminho pode também ser feita a partir do próprio usuário. Para tanto, arquiteturas que ofereçam ao usuário a oportunidade de escolher qual o caminho seguido pelo seu tráfego no nível de ASes devem ser desenvolvidas. As propostas nessa área podem ser classificadas de duas maneiras: orientadas a agentes ou orientadas a usuários.

As propostas orientadas a agentes estão na direção oposta à premissa fim-a-fim da Internet. A característica de inteligência nas bordas, por um lado, aumenta a simplicidade e flexibilidade da rede [Moreira et al. 2009]. Entretanto, por outro lado, dificulta o diagnóstico de falhas e implica em configurações manuais. Essas últimas consequências reduzem o desempenho da rede já que podem gerar configurações erradas e levar ao aumento do tempo de recuperação da rede. A inserção de agentes inteligentes na rede tem por objetivo reduzir os problemas acarretados por configurações manuais. Entretanto, o emprego dos agentes não deve prejudicar um dos principais pilares da Internet que é a simplicidade do núcleo [Clark et al. 2003].

O papel dos agentes na Internet é aumentar a autonomia da rede ao ponto de torná-la o mais independente possível da intervenção humana. Atualmente, o crescimento do número de nós, de usuários e da demanda por conectividade e banda passante têm tornado o gerenciamento das redes mais e mais complexo. O gerenciamento humano, por conseguinte, pode se tornar um limitante para o crescimento da Internet. O conceito de redes autonômicas defende o uso de técnicas que permitam à rede conhecer o contexto que está

inserida e o que lhe é solicitada. Assim, torna-se possível realizar ações sem a intervenção humana. As redes autonômicas também devem ter uma visão em alto nível dos objetivos da rede e das suas limitações para tomar decisões de configuração. Por fim, os agentes devem relatar o seu desempenho em alto nível para usuários ou administradores.

O conhecimento do tráfego encaminhado e dos requisitos dos usuários permite que os agentes tomem decisões no nível do roteamento. Tais decisões não necessariamente são regidas por resultados de algoritmos determinísticos. Isso ocorre porque o ambiente no qual as decisões são tomadas é altamente dinâmico e sujeito a conflito de interesses e a informações incompletas. Nesse caso, é importante manter uma base de dados que construa, reconcilie e mantenha os muitos aspectos de uma visão em alto nível do comportamento da Internet. Essa base de dados é separada em um novo plano denominado Plano de Conhecimento (*Knowledge Plane*) por Clark *et al.* [Clark et al. 2003]. O Plano de Conhecimento deve ser capaz de prover serviços aos outros elementos da rede como, por exemplo, qual a melhor configuração em um dado momento de operação da rede. No caso do roteamento, o Plano de Conhecimento é capaz de alterar ou estabelecer caminhos conforme requisitos de aplicações. Um dos desafios do Plano de Conhecimento é mensurar o quanto de informações ele deve possuir e qual o alcance delas. Especialmente no roteamento, projetar um Plano de Conhecimento que possua informações globais da Internet não é viável dada a massa de informações necessárias. Logo, o Plano de Conhecimento deve conhecer o tipo de informação que é mais útil em uma dada circunstância e deve usar técnicas distribuídas escaláveis para filtrar as observações conforme os interesses dos usuários.

Nas propostas orientadas a usuários, estes possuem maior liberdade para escolher o tipo de caminho pelo qual os seus pacotes são encaminhados. A arquitetura NIRA (*New Internet Routing Architecture*) [Yang et al. 2007, Yang 2003] é a principal representante desse tipo de propostas. A arquitetura NIRA tem como um dos seus objetivos estimular a competição entre os ISPs. Esse é o motivo pelo qual a arquitetura NIRA propõe que os usuários tenham oportunidade de escolher os caminhos no nível de ASes e não no nível de roteadores internos a um AS. A competição entre ISPs pode levar à redução dos custos do acesso à Internet para os usuários bem como estimular a introdução de valor agregado aos serviços. Dentre possíveis valores agregados poderia estar, por exemplo, caminhos com qualidade de serviço fim-a-fim.

A maneira mais trivial para os usuários definirem o caminho percorrido pelos pacotes é o uso de roteamento pela fonte. Entretanto, o roteamento pela fonte aumenta o tamanho dos cabeçalhos. A arquitetura NIRA utiliza um esquema de roteamento pela fonte que não introduz o caminho explicitamente no cabeçalho do pacote. Na NIRA, os endereços das estações são a concatenação do identificador da estação com os identificadores de todos os ASes até o núcleo da Internet. A parte do endereço que identifica os ASes deve ser uma concatenação dos prefixos de todos os domínios no caminho até os ASes de núcleo. A Figura 2.27 ilustra um exemplo, supondo que haja apenas um domínio entre a rede de acesso e o núcleo, o endereço da estação A é 1.1.1.1, que é uma concatenação do identificador da estação (1), do identificador do domínio intermediário (1.1.1.0/24) e do núcleo (1.1.0.0/16). Assim, o caminho do pacote fica definido até o núcleo da Internet a partir do endereço da estação. Em uma comunicação fim-a-fim, a maneira como os endereços de origem e de destino dos pacotes são organizados é suficiente

para que o caminho seja totalmente definido. Apenas os ASes de núcleo precisam executar um protocolo de roteamento para encaminhar os pacotes recebidos. Na Figura 2.27, o protocolo de roteamento define o melhor caminho a ser seguido pelos pacotes entre o roteador R_A e R_B, por exemplo. Como o número de ASes no núcleo é pequeno em relação ao número de ASes da Internet, esse protocolo não enfrenta problemas de escalabilidade.

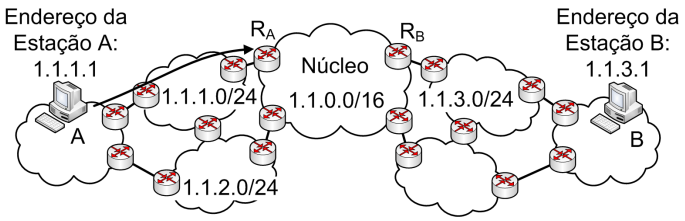


Figura 2.27. Estrutura de endereços utilizados na NIRA.

O encaminhamento de pacotes baseado tanto no endereço de origem quanto no destino dos pacotes é diferente do encaminhamento de pacotes da Internet que é baseado apenas no endereço de destino. Entretanto, a definição do caminho nos endereços dos pacotes ainda não é suficiente para prover liberdade de escolha aos usuários. Assim, é necessário um procedimento de descobrimento de rotas para que os usuários possam escolher, dentre os caminhos possíveis, o melhor dependendo dos seus interesses. A NIRA utiliza mecanismos de descoberta de caminhos antes do envio de pacotes. Para tal, a NIRA define dois protocolos para auxiliar a descoberta de caminhos: o TIPP (*Topology Information Propagation Protocol*) e o NRRS (*Name-to-Route Resolution Service*). O primeiro protocolo propaga aos usuários suas informações de endereços intradomínio levando em consideração possíveis acordos entre ASes. As informações correspondem ao caminho desde a estação até os ASes de núcleo. Na Figura 2.27, a estação A poderia escolher entre o endereço 1.1.1.1 e o endereço 1.1.2.1. Já o protocolo NRRS define um serviço de resolução de nomes para que cada estação conheça o endereço das outras estações da rede. Assim, a estação A descobre o endereço da estação B, 1.1.3.1, como visto na Figura 2.27. Os endereços de origem e destino são armazenados em memória cache para evitar buscas consecutivas pelos mesmos endereços, exceto em caso de falhas.

Outra proposta orientada a usuários é o *Pathlet Routing* [Godfrey et al. 2009]. No *Pathlet Routing*, cada ISP anuncia fragmentos de caminhos, denominados *Pathlets*, que podem ser concatenados pelo usuário para formar um caminho fim-a-fim. Um *Pathlet* é definido como uma sequência de nós considerados virtuais. Tais nós podem estar associados a todas as rotas conhecidas pelos roteadores da rede que pertencem, ou no caso mais simples, a apenas as rotas conhecidas por um único roteador. O caminho de um pacote é definido na fonte e inserido no cabeçalho do pacote. Logo, o *Pathlet Routing* utiliza roteamento pela fonte. Cada *Pathlet* é identificado por um identificador de encaminhamento (*Forwarding IDentifier* - FID) cujo significado é local ao nó virtual de origem do *Pathlet*. Uma vez que o roteamento é pela fonte, é importante apenas para o nó de origem do *Pathlet* conhecer para que nó encaminhar os pacotes recebidos. Ao encaminhar um pacote, o roteador altera a sequência de identificadores. A Figura 2.28 ilustra essas alterações. Na figura, cada seta pontilhada representa um *Pathlet* e os retângulos abaixo

de cada roteador representam os identificadores dos *Pathlets* contidos no cabeçalho do pacote naquele roteador. O pacote é originado no roteador A e destinado ao roteador E. Note que a cada salto os *Pathlets* atravessados são removidos do cabeçalho (ex. entre os roteadores A e B) e novos *Pathlets* são inseridos no cabeçalho do pacote caso um *Pathlet* atravessado seja composto por múltiplos saltos (ex. entre os roteadores B e C já que o *Pathlet* 2 é composto pelos *Pathlets* 7 e 1). *Pathlets* compostos são formados assim que um determinado roteador aprende múltiplos *Pathlets* consecutivos.

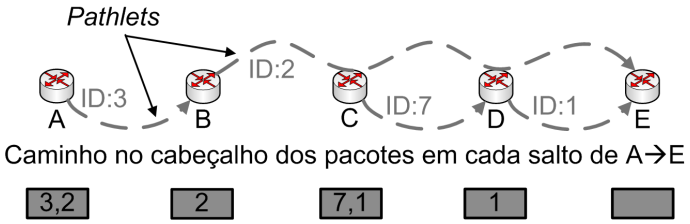


Figura 2.28. *Pathlet Routing*.

No *Pathlet Routing*, um determinado nó da rede dissemina os *Pathlets* conhecidos que traduzem as suas políticas. A disseminação é realizada através de vetores de caminhos no BGP. Os *Pathlets* recebidos pelos usuários são utilizados na escolha dos caminhos. Uma das vantagens do uso dos *Pathlets* é a economia no tamanho dos cabeçalhos. Na Figura 2.28, o roteamento pela fonte precisaria de cinco endereços IP enquanto com o uso do *Pathlet* precisa de no máximo dois identificadores (FIDs).

Uma questão importante é como incentivar os ISPs a disponibilizarem aos usuários a liberdade de escolha de caminhos. O modelo atual da Internet é baseado em acordos comerciais entre os provedores. Portanto, a manutenção desse modelo representa uma restrição à possibilidade de escolha dos usuários. A mudança do modelo, entretanto, deve ser atraente também para os ISPs. Alguns modelos de negócios para os provedores incluem acordos entre os usuários e os seus provedores de acesso e entre os usuários e os provedores de todo o caminho até o núcleo. Na primeira opção, os provedores conectados diretamente aos usuários fazem acordos com os provedores vizinhos para oferecerem diferentes possibilidades de caminhos para a escolha dos usuários. Os provedores tarifam os usuários através do monitoramento dos caminhos que os usuários escolheram. Já na segunda opção, os usuários têm a possibilidade de escolher todos os ASes no caminho. Embora essa última opção ofereça mais liberdade, ela pode sofrer problemas de adoção, visto que qualquer provedor de serviço segue algum tipo de política com os seus vizinhos.

Uma dificuldade que ambos os tipos de proposta, orientada a agentes e orientada a usuários, enfrenta é a aquisição de informações da rede. Tais informações são importantes para que tanto os agentes quanto os usuários possam embasar suas decisões. Por exemplo, aplicações como VoIP possuem restrições de atraso fim-a-fim. Essa métrica não é simples de ser obtida visto que necessita de sincronismo entre relógios. Além da obtenção de métricas, outro desafio é consolidar as informações e representar o estado atual da rede a partir dessas informações. O estado da rede deve retratar de maneira mais fiel possível dependendo das necessidades das aplicações. A arquitetura CONMan (*Complexity Oblivious Network Management*) [Ballani e Francis 2007] propõe uma interface para simplificar a obtenção de informações de protocolos. Os autores argumentam que uma

das grandes dificuldades para o gerenciamento de redes é que os protocolos e dispositivos exibem muitos detalhes de sua implementação, o que dificulta a obtenção dos dados. A arquitetura CONMan, portanto, propõe uma interface que inclui o mínimo necessário de informações específicas de protocolos e outros dispositivos para simplificar a obtenção de dados, e assim, melhorar o desempenho da rede a partir de um gerenciamento mais efetivo.

2.4.7. OpenFlow e a solução comutada

Atualmente, o Ethernet é uma tecnologia consolidada para redes locais e metropolitanas. O Ethernet é uma solução comutada e, portanto, não pode ser aplicada diretamente na Internet que é roteada. A simplicidade do Ethernet e a facilidade de configuração de redes, entretanto, vêm estimulando o emprego dessa tecnologia em redes de maior escala. Para isso, procedimentos como o de inundação de controle, utilizado para localização de estações, configuração automática de endereços através de DHCP (*Dynamic Host Configuration Protocol*) e resolução ARP (*Address Resolution Protocol*); e o emprego de árvores de espalhamento (*spanning tree*) para comunicação entre os nós devem ser revistos. Além disso, em uma rede comutada, cada comutador armazena a localização de cada destino na rede. Para contornar essas características e tornar a solução comutada mais escalável, trabalhos na área propõem soluções híbridas como o emprego de roteadores executando o IP para interconectar redes Ethernet e o emprego de VLANs (*Virtual LANs*) [Kim et al. 2008]. Outras propostas, como o Seattle [Kim et al. 2008], reduzem o número de estados por comutador e ainda evitam inundações. Para isso, o Seattle utiliza um sistema de localização de estações através de DHTs, e assim, dispensa o armazenamento de informações globais em cada um dos nós da rede.

Um das soluções comutadas que vem se destacando é o OpenFlow. O OpenFlow [McKeown et al. 2008, Mateo 2009] possibilita a programabilidade de elementos de rede, sejam eles comutadores ou roteadores. Essa característica permite programar redes e, como consequência, realizar experimentos isolados simultâneos para testes de novas propostas de roteamento e até mesmo alternativas ao IP. O OpenFlow permite que fluxos sejam definidos, bem como os caminhos seguidos por cada fluxo, sem interferir nos outros fluxos da rede. Os caminhos podem ser obtidos a partir de métodos oferecidos pelo OpenFlow. Tais métodos definem políticas para busca automática de caminhos que atendam requisitos desejados como largura de banda disponível e atraso restrito.

Uma rede convencional é composta por enlaces, elementos de processamento de pacotes (comutadores/roteadores) e estações finais. Nessas redes, cada elemento de processamento agrega funções de encaminhamento de pacotes e decisões de controle. No OpenFlow, entretanto, essas duas funções são separadas. A função de encaminhamento de pacote continua nos comutadores/roteadores. Porém, as decisões de controle são atribuídas a um novo elemento de rede, chamado de controlador. Nesse sentido, o OpenFlow altera a arquitetura convencional de redes ao introduzir o conceito de controladores. Os controladores separados foram propostos para que a programabilidade dos comutadores possa ser realizada externamente ao equipamento para que não haja dependência do fabricante. Essa característica facilita a adoção da tecnologia. Para que isso seja possível, o OpenFlow define uma interface e um protocolo para comunicação entre controladores e comutadores. Esse protocolo define mensagens para modificação das tabelas de enca-

minhamento, recepção e envio de pacotes, busca de estatísticas da rede e informações do equipamento. Essas mensagens são utilizadas para realizar determinadas funções, dentre as quais a mais importante para prover programabilidade é a modificação das tabelas de encaminhamento. A arquitetura de uma rede OpenFow é ilustrada na Figura 2.29.

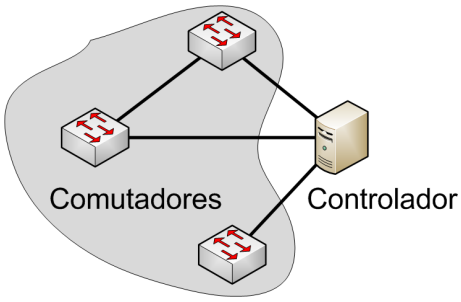


Figura 2.29. Arquitetura de uma rede OpenFlow.

A programabilidade do OpenFlow é obtida através da possibilidade de alterar a tabela de encaminhamento, que no caso da comutação, é chamada de tabela de fluxos. A tabela de fluxos define a ação associada a cada fluxo e estabelece o tipo de processamento a ser realizado sobre aquele fluxo. Cada fluxo é identificado na tabela a partir de informações contidas no seu cabeçalho. Informações como endereços MAC, endereços IP, portas TCP etc. são utilizadas para identificação. Uma vez identificado o fluxo, o OpenFlow executa a ação relacionada a ele como descrito na própria tabela. A presença de informação além de informações de camada de enlace pode ser usada por um comutador para que ele se comporte como um roteador ou como um equipamento de camada acima da camada de transporte. Da mesma maneira, um fluxo pode ser apenas identificado pelo cabeçalho da camada de enlace. Isso facilita o teste de soluções que não utilizam o IP como protocolo de rede. O OpenFlow não requer que o cabeçalho dos pacotes sigam um determinado formato. Ele requer apenas que o formato utilizado possa ser identificado na tabela de fluxos. Essa característica facilita a experimentação de novos protocolos para a Internet do Futuro.

Os pacotes recebidos que não possuem entradas correspondentes na tabela de fluxos são enviados ao controlador que decide a ação a ser realizada nesse pacote. Portanto, sempre que um pacote recebido não encontrar uma porta comutada de saída já estabelecida, ele é encaminhado para o controlador através do canal de comunicação. Esse canal utiliza SSL (*Secure Sockets Layer*) para garantir autenticação e confidencialidade, já que o protocolo definido pelo OpenFlow faz mudanças na tabela de fluxos dos comutadores. Após a análise do controlador, uma entrada é adicionada ou não na tabela de fluxos do comutador correspondente. Caso uma entrada seja adicionada, os próximos pacotes são encaminhados sem a necessidade de passar primeiro pelo controlador.

Uma rede OpenFlow é composta de um ou mais comutadores OpenFlow, um ou mais controladores e um canal seguro para comunicação. Cada comutador OpenFlow possui sua tabela de fluxos enquanto o canal seguro é usado para comunicação entre os comutadores e os controladores. Normalmente, em uma rede OpenFlow há comutadores e apenas um controlador. Como consequência, o OpenFlow possui controle centralizado. Essa centralização representa um compromisso entre escalabilidade e facilidade

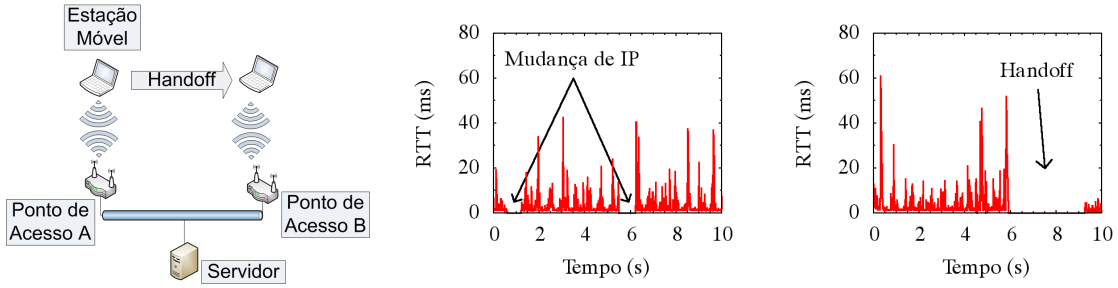
de gerenciamento. Por um lado, a centralização requer que todas as decisões de controle e gerenciamento da rede sejam executadas por apenas um elemento. Esse elemento pode representar um ponto de vulnerabilidade e um entrave dependendo do número de requisições realizadas. A proposta original do OpenFlow [McKeown et al. 2008] deixa essa limitação clara já que propõe o uso dessa técnica de comutação em redes de *campi* universitários. Por outro lado, porém, facilita a implementação de algoritmos para escolher quais as ações apropriadas para os fluxos correntes. Uma das vantagens do OpenFlow é o isolamento de redes concorrentes. Nesse caso, mais de um controlador é usado, um para cada rede. O OpenFlow oferece uma camada de abstração denominada FlowVisor [Sherwood et al. 2010] para compartilhamento de recursos físicos entre diferentes redes isoladas.

Dentre as ações que podem ser realizadas pelo OpenFlow estão o encaminhamento de pacotes, remoção de pacotes e a criação de fluxos. Os pacotes de um fluxo podem ser encaminhados para uma determinada porta conforme requisitos desejados. Essa facilidade permite que os pacotes sejam roteados através da rede. Além disso, outra ação relacionada ao encaminhamento de pacotes pode definir que todos os pacotes de um fluxo devem primeiro ser encaminhados para o controlador. Apesar dessa opção reduzir a velocidade de encaminhamento de pacote, ela pode facilitar operações dinâmicas visto que é somente possível executar algoritmos no controlador. Outra ação que pode ser definida pelo OpenFlow é o descarte de alguns ou todos os pacotes relacionados com um determinado fluxo. Já a criação de fluxos pode ocorrer caso haja necessidade de realizar engenharia de tráfego. As ações podem ser usadas para gerenciar o tráfego da rede. É importante notar que todas as ações são definidas pelo controlador. Controladores mais sofisticados como o NOX [Gude et al. 2008] oferecem interfaces para o desenvolvimento de aplicações para gerenciamento ou para obtenção de informações da rede. Dentre possíveis aplicações estão a manutenção de conexões mesmo após o deslocamento de usuários móveis. Nessa aplicação, os pontos de acesso sem-fio monitoram os usuários conectados. Uma vez que um usuário mude de ponto de acesso, o ponto de acesso relata ao controlador que realiza ações para manter as conexões estabelecidas pelo usuário móvel. Essas ações correspondem à mudança do caminho seguido pelos pacotes destinados a esse usuário. Logo, as tabelas de fluxo são ajustadas para que o fluxo correspondente seja encaminhado até o novo ponto de acesso no qual o usuário está conectado.

2.5. Resultados Experimentais

Esta seção apresenta resultados experimentais obtidos com um protótipo realizado para separação de identificação e localização. O objetivo dos experimentos é verificar o desempenho de um dos novos protocolos para mobilidade na Internet a partir de mudanças do ponto de interconexão com a rede cabeada. O protocolo escolhido foi o HIP, descrito na Seção 2.4.1. O cenário dos experimentos, ilustrado pela Figura 2.30(a), é composto de uma estação móvel, um servidor na rede cabeada e dois pontos de acesso sem-fio. A estação móvel é um Laptop IBM Thinkpad Intel Pentium M 1.7 GHz com 512 MB. Já o servidor utilizado é um computador de mesa Intel Core2 Duo 2.4 GHz com 2 GB de memória RAM. Por fim, os pontos de acesso são dois roteadores Linksys WRT54G e WRT350N. A ferramenta utilizada para implementar os protocolos contidos no HIP foi a ferramenta OpenHip versão 0.7 [OpenHIP 2009].

O objetivo do primeiro experimento é verificar o tempo mínimo necessário para o HIP atualizar o endereço IP de uma estação móvel. Para isso, a estação móvel permanece parada e o endereço IP é alterado manualmente duas vezes. O tempo de atualização é, basicamente, o tempo que uma estação sem-fio executando o HIP precisa para notificar as outras estações da rede da sua mudança de endereço. Nesse experimento, a estação móvel permanece parada enviando mensagens ICMP de requisição (`ping`) para um servidor da rede cabeada através do ponto de acesso A. Em um dado momento a estação móvel modifica seu endereço IP para forçar o procedimento de atualização de endereços do HIP. Durante esse período, os pacotes ICMP de resposta enviados para a estação móvel são perdidos já que o endereço IP da estação mudou. A Figura 2.30(b) ilustra o tempo de ida-e-volta (*Round Trip Time* - RTT) das mensagens ICMP enviadas durante 10 segundos de experimento. A figura ilustra os momentos em que as duas alterações de endereço IP foram executadas. Note que após as alterações, o HIP leva, na média, 677 milissegundos para restabelecer a comunicação. Esse tempo é decorrente principalmente da implementação do OpenHip que processa mensagens do HIP apenas periodicamente.



(a) Cenário de teste. (b) Atualização de endereços IP. (c) *Handoff* da estação móvel.

Figura 2.30. Experimentos práticos.

O segundo experimento avalia o impacto da mudança dos endereços IP na transferência de dados utilizando o HIP. Nesse experimento, a estação móvel se desloca do ponto de acesso sem-fio A até o ponto de acesso B, realizando assim um processo de *handoff*. Nesse teste, a estação móvel também envia mensagens de requisição ICMP e espera os pacotes de resposta do servidor na rede cabeada. Durante o processo de *handoff*, a mudança dos pontos de acesso é realizada quando a intensidade do sinal do ponto de acesso B supera a intensidade do sinal do ponto de acesso A. Ao detectar esse evento, a estação móvel conecta-se ao ponto de acesso B e troca de endereço IP. A Figura 2.30(c) mostra que o processo de *handoff* e a troca de mensagens de atualização do HIP duraram 3,354 segundos. Esse tempo é decorrência da troca não instantânea de pontos de acesso, o que resulta em perdas de mensagens de atualização e, consequentemente, maior tempo até o processo de atualização de endereço ser concluído pelo HIP.

Os resultados demonstram que a mobilidade na Internet é viável, mas ainda precisa de ajustes para se tornar transparente aos usuários.

2.6. Considerações Finais

Ao longo dos anos, a Internet vem sofrendo adaptações para atender demandas não previstas originalmente. Tais mudanças são observadas na camada de interconexão

de redes através de novas propostas para substituição e adaptação do IP e dos principais protocolos de roteamento intra e interdomínio. A principal dificuldade dessas propostas é que elas são tentativas de adequar a Internet a uma realidade em constante evolução. A outra possibilidade é o rompimento completo com o projeto inicial da Internet e a adoção de uma nova proposta totalmente inovadora. A experiência adquirida com os anos de operação da Internet poderia ser utilizada em uma proposta evolutiva para a Internet do Futuro. A primeira opção vem sendo adotada pelos provedores de serviço já que implica em mudanças gerenciáveis. Entretanto, por se tratarem de mudanças de curto prazo, até hoje não resolveram por completo os problemas da Internet. Já a segunda opção pode exigir maiores dispêndios financeiros dos provedores de serviço por ser mais radical, em compensação pode ser mais vantajosa no longo prazo. Enquanto esse impasse persiste, demandas emergentes como a mobilidade das estações, os múltiplos domicílios, os múltiplos caminhos e a escalabilidade dos roteadores continuam sem uma solução definitiva satisfatória.

Este minicurso apresentou propostas tanto adaptativas quanto radicais para a interconexão de redes na Internet do Futuro. Algumas propostas se mostraram mais preocupadas com a questão da implementação gradual embora ainda impliquem em mudanças na rede. A lição aprendida é que o problema é complexo e talvez não seja possível atender todas as demandas emergentes e ainda ser escalável e financeiramente factível. Algumas das demandas podem continuar como casos a parte mesmo naquela que será a Internet do Futuro. Uma nova arquitetura que mantenha as mesmas características que fizeram da Internet um dos maiores sucessos do século XX e, ao mesmo tempo, que atenda as novas demandas é um problema em aberto para os próximos anos e ainda despertará grande interesse na comunidade científica.

Agradecimentos

Este trabalho utilizou recursos da CAPES, CNPq, FAPERJ, FUJB, FUNTTEL e FINEP.

Referências

[Alves et al. 2009] Alves, R. S., Campbell, I. V., Couto, R. S., Campista, M. E. M., Moraes, I. M., Rubinstein, M. G., Costa, L. H. M. K., Duarte, O. C. M. B. e Abdalla, M. (2009). *Redes Veiculares: Princípios, Aplicações e Desafios*, capítulo 5, páginas 199–254. Minicursos do Simpósio Brasileiro de Redes de Computadores (SBRC). Sociedade Brasileira de Computação (SBC).

[Andersen et al. 2008] Andersen, D. G., Balakrishnan, H., Feamster, N., Koponen, T., Moon, D. e Shenker, S. (2008). Accountable Internet Protocol (AIP). Em *ACM SIGCOMM*, páginas 339–350.

[Andersen et al. 2001] Andersen, D. G., Balakrishnan, H., Kaashoek, M. F. e Morris, R. (2001). Resilient Overlay Networks. *ACM SIGOPS Operating Systems Review*, 35(5):131–145.

[Ballani e Francis 2007] Ballani, H. e Francis, P. (2007). CONMan: a step towards network manageability. *ACM SIGCOMM Computer Communication Review*,

37(4):205–216.

[Ballani et al. 2008] Ballani, H., Francis, P., Cao, T. e Wang, J. (2008). ViAggre: Making routers last longer! Em *Workshop on Hot Topics in Networks (HotNets-VII)*, páginas 1–6.

[Ballani et al. 2009] Ballani, H., Francis, P., Cao, T. e Wang, J. (2009). Making routers last longer with ViAggre. Em *Symposium on Networked Systems Design and Implementation (NSDI)*, páginas 453–466.

[Brim et al. 2008] Brim, S., Chiappa, N., Farinacci, D., Fuller, V., Lewis, D. e Meyer, D. (2008). LISP-CONS: A Content distribution Overlay Network Service for LISP. IETF Network Working Group Internet-Draft (Trabalho em andamento).

[Caesar et al. 2006a] Caesar, M., Castro, M., Nightingale, E. B., O'Shea, G. e Rowstron, A. (2006a). Virtual Ring Routing: network routing inspired by DHTs. Em *ACM SIGCOMM*, páginas 351–362.

[Caesar et al. 2006b] Caesar, M., Condie, T., Kannan, J., Lakshminarayanan, K., Stoica, I. e Shenker, S. (2006b). ROFL: Routing On Flat Labels. Em *ACM SIGCOMM*, páginas 363–374.

[Caida 2010] Caida (2010). Visualizing IPv4 and IPv6 Internet topology at a macroscopic scale. Acessado 10/03/10 em http://www.caida.org/research/topology/as_core_network/.

[CIDR 2009] CIDR (2009). Aggregation summary. Acessado 18/12/2009 em http://www.cidr-report.org/as2.0/.

[CIDR 2010] CIDR (2010). Active BGP entries (FIB). Acessado 16/03/2010 em http://www.cidr-report.org/as2.0/.

[Clark et al. 2004] Clark, D., Braden, R., Sollins, K., Wroclawski, J., Katabi, D., Kulik, J., Yang, X., Faber, T., Falk, A., Pingali, V., Handley, M. e Chiappa, N. (2004). New Arch: Future generation Internet architecture. Relatório técnico, USC Information Sciences Institute Computer Networks Division, MIT Laboratory for Computer Science and International Computer Science Institute (ICSI).

[Clark et al. 2003] Clark, D. D., Partridge, C., Ramming, J. C. e Wroclawski, J. T. (2003). A knowledge plane for the Internet. Em *ACM SIGCOMM*, páginas 3–10.

[Clausen et al. 2001] Clausen, T., Jacquet, P., Laouiti, A., Muhlethaler, P., Qayyum, A. e Viennot, L. (2001). Optimized link state routing protocol. Em *IEEE International Multi Topic Conference (INMIC)*, páginas 62–68.

[Costa et al. 2006] Costa, L. H. M. K., Fdida, S. e Duarte, O. C. M. B. (2006). Incremental service deployment using the Hop By Hop Multicast Routing Protocol. *IEEE/ACM Transactions on Networking*, 14(3):543–556.

[Crowcroft 2008] Crowcroft, J. (2008). Toward a network architecture that does everything. *Communications of ACM*, 51(1):74–77.

[Deering e Hinden 1998] Deering, S. e Hinden, R. (1998). Internet Protocol, version 6 (IPv6) specification. IETF Network Working Group RFC 2460.

[Deering 1988] Deering, S. E. (1988). Multicast routing in internetworks and extended LANs. Em *ACM SIGCOMM*, páginas 55–64.

[Devarapalli et al. 2005] Devarapalli, V., Wakikawa, R., Petrescu, A. e Thubert, P. (2005). Network Mobility (NEMO) basic support protocol. IETF Network Working Group RFC 3963.

[Farinacci et al. 2009a] Farinacci, D., Fuller, V., Meyer, D. e Lewis, D. (2009a). LISP ALternative Topology (LISP+ALT). IETF Network Working Group Internet-Draft (Trabalho em andamento).

[Farinacci et al. 2009b] Farinacci, D., Fuller, V., Oran, D. e Meyer, D. (2009b). Locator/ID separation protocol (LISP). IETF Network Working Group Internet-Draft (Trabalho em andamento).

[Feldmann et al. 2009] Feldmann, A., Cittadini, L., Mühlbauer, W., Bush, R. e Maennel, O. (2009). HAIR: Hierarchical Architecture for Internet Routing. Em *ACM Workshop on Re-Architecting the Internet (ReArch)*, páginas 43–48.

[Ganesan et al. 2004] Ganesan, P., Gummadi, K. e Garcia-Molina, H. (2004). Canon in G major: Designing DHTs with hierarchical structure. Em *International Conference on Distributed Computing Systems (ICDCS)*, páginas 263–272.

[Godfrey et al. 2009] Godfrey, P. B., Ganichev, I., Shenker, S. e Stoica, I. (2009). Pathlet routing. Em *ACM SIGCOMM*, páginas 111–122.

[Gude et al. 2008] Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N. e Shenker, S. (2008). NOX: Towards an operating system for networks. *ACM SIGCOMM Computer Communication Review*, 38(3):105–110.

[Hanbali et al. 2005] Hanbali, A. A., Altman, E. e Nain, P. (2005). Survey of TCP over ad hoc networks. *IEEE Communications Surveys & Tutorials*, 7(3):22–36.

[He e Rexford 2008] He, J. e Rexford, J. (2008). Towards Internet-wide multipath routing. *IEEE Network*, 22(2):16–21.

[Hinden e Sheltzer 1982] Hinden, R. e Sheltzer, A. (1982). The DARPA Internet gateway. IETF Network Working Group RFC 823.

[History Museum 2010] History Museum, C. (2010). Internet history. Acessado em http://www.computerhistory.org.

[Iannone e Bonaventure 2007] Iannone, L. e Bonaventure, O. (2007). On the cost of caching locator/ID mappings. Em *ACM CoNEXT*, páginas 1–12.

[Internet World Stats 2010] Internet World Stats (2010). World Internet users and population stats. Acessado em http://www.internetworldstats.com/stats.htm.

[Jen et al. 2009] Jen, D., Meisel, M., Massey, D., Wang, L., Zhang, B. e Zhang, L. (2009). APT: A Practical Tunneling architecture for routing scalability. Relatório técnico, UCLA.

[Jen et al. 2008] Jen, D., Meisel, M., Yan, H., Massey, D., Wang, L., Zhang, B. e Zhang, L. (2008). Towards a new Internet routing architecture: Arguments for separating edges from transit core. Em *ACM Workshop on Hot Topics in Networks (HotNets)*, páginas 1–6.

[Johnson et al. 2004] Johnson, D., Perkins, C. e Arkko, J. (2004). Mobility support in IPv6. IETF Network Working Group RFC 3775.

[Kaur et al. 2003] Kaur, H. T., Kalyanaraman, S., Weiss, A., Kanwar, S. e Gandhi, A. (2003). BANANAS: an evolutionary framework for explicit and multipath routing in the Internet. Em *ACM SIGCOMM workshop on Future Directions in Network Architecture (FDNA)*, páginas 277–288.

[Kim et al. 2009] Kim, C., Caesar, M., Gerber, A. e Rexford, J. (2009). Revisiting route caching: The world should be flat. Em *International Conference on Passive and Active Network Measurement (PAM)*, páginas 3–12.

[Kim et al. 2008] Kim, C., Caesar, M. e Rexford, J. (2008). Floodless in SEATTLE: a Scalable Ethernet Architecture for Large Enterprises. Em *ACM SIGCOMM*, páginas 3–14.

[Krioukov et al. 2007] Krioukov, D., kc claffy, Fall, K. e Brady, A. (2007). On compact routing for the Internet. *ACM SIGCOMM Computer Communication Review*, 37(3):41–52.

[Lear 2010] Lear, E. (2010). NERD: A Not-so-novel EID to RLOC Database. IETF Network Working Group Internet-Draft (Trabalho em andamento).

[Lougheed e Rekhter 1989] Lougheed, K. e Rekhter, Y. (1989). A Border Gateway Protocol (BGP). IETF Network Working Group RFC 1105.

[Luo et al. 2009] Luo, H., Qin, Y. e Zhang, H. (2009). A DHT-based identifier-to-locator mapping approach for a scalable Internet. *IEEE Transactions on Parallel and Distributed Systems*, 20(12):1790–1802.

[Massey et al. 2007] Massey, D., Wang, L., Zhang, B. e Zhang, L. (2007). A scalable routing system design for future Internet. Em *ACM SIGCOMM IPv6 and the Future of the Internet Workshop*, páginas 1–6.

[Mateo 2009] Mateo, M. P. (2009). OpenFlow switching performance. Dissertação de mestrado, Politecnico Di Torino, Torino, Itália.

[McCarthy et al. 2008a] McCarthy, B., Edwards, C. e Dunmore, M. (2008a). Using NEMO to extend the functionality of MANETs. Em *IEEE International Conference on Communications (ICC)*, páginas 455–460.

[McCarthy et al. 2009] McCarthy, B., Edwards, C. e Dunmore, M. (2009). Using NEMO to support the global reachability of MANET nodes. Em *IEEE Conference on Computer Communications (INFOCOM)*, páginas 2097–2105.

[McCarthy et al. 2008b] McCarthy, B., Jakeman, M., Edwards, C. e Thubert, P. (2008b). Protocols to efficiently support nested NEMO (NEMO+). Em *International workshop on Mobility in the evolving Internet Architecture (MobiArch)*, páginas 43–48.

[McKeown et al. 2008] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S. e Turner, J. (2008). Openflow: Enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74.

[Meyer et al. 2007] Meyer, D., Zhang, L. e Fall, K. (2007). Report from the IAB Workshop on Routing and Addressing. Acessado em http://tools.ietf.org/html/draft-iab-raws-report-02.txt.

[Mills 1984] Mills, D. L. (1984). Exterior Gateway Protocol formal specification. IETF Network Working Group RFC 904.

[Moreira et al. 2009] Moreira, M. D. D., Fernandes, N. C., Costa, L. H. M. K. e Duarte, O. C. M. B. (2009). *Internet do Futuro: Um Novo Horizonte*, capítulo 1, páginas 1–59. Minicursos do Simpósio Brasileiro de Redes de Computadores (SBRC). Sociedade Brasileira de Computação (SBC).

[Moskowitz et al. 2008] Moskowitz, R., Nikander, P., Jokela, P. e Henderson, T. (2008). Host Identity Protocol. IETF Network Working Group RFC 5201.

[Motiwala et al. 2008] Motiwala, M., Elmore, M., Feamster, N. e Vempala, S. (2008). Path splicing. Em *ACM SIGCOMM*, páginas 27–38.

[Nikander et al. 2008] Nikander, P., Henderson, T., Vogt, C. e Arkko, J. (2008). End-host mobility and multihoming with the Host Identity Protocol. IETF Network Working Group RFC 5206.

[Nordmark e Bagnulo 2009] Nordmark, E. e Bagnulo, M. (2009). Shim6: Level 3 multihoming Shim protocol for IPv6. IETF Network Working Group RFC 5533.

[OpenHIP 2009] OpenHIP (2009). Host Identity Protocol (HIP). Acessado em http://www.openhip.org/.

[Perera et al. 2004] Perera, E., Sivaraman, V. e Seneviratne, A. (2004). Survey on network mobility support. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(2):7–19.

[Perkins 2002] Perkins, C. (2002). IP mobility support for IPv4. IETF Network Working Group RFC 3344.

[Ratnasamy et al. 2005] Ratnasamy, S., Shenker, S. e McCanne, S. (2005). Towards an evolvable Internet architecture. *ACM SIGCOMM Computer Communication Review*, 35(4):313–324.

[Rekhter e Li 1995] Rekhter, Y. e Li, T. (1995). A Border Gateway Protocol 4 (BGP-4). IETF Network Working Group RFC 1771.

[Rekhter et al. 2006] Rekhter, Y., Li, T. e Hares, S. (2006). A Border Gateway Protocol 4 (BGP-4). IETF Network Working Group RFC 4271.

[Sabeur et al. 2006] Sabeur, M., Jouaber, B. e Zeghlache, D. (2006). Light-NEMO+: Route optimzation for light-NEMO solution. Em *IEEE International Conference on Networks (ICON)*, páginas 1–6.

[Saucez et al. 2009] Saucez, D., Iannone, L. e Bonaventure, O. (2009). OpenLISP: An open source implementation of the locator/ID separation protocol. Em *ACM SIGCOMM Demos Session*, páginas 1–2.

[Sherwood et al. 2010] Sherwood, R., Chan, M., Covington, A., Gibb, G., Flajslik, M., Handigol, N., Huang, T.-Y., Kazemian, P., Kobayashi, M., Naous, J., Seetharaman, S., Underhill, D., Yabe, T., Yap, K.-K., Yiakoumis, Y., Zeng, H., Appenzeller, G., Johari, R., McKeown, N. e Parulkar, G. (2010). Carving research slices out of your production networks with OpenFlow. *ACM SIGCOMM Computer Communication Review*, 40(1):129–130.

[Stoica et al. 2004] Stoica, I., Adkins, D., Zhuang, S., Shenker, S. e Surana, S. (2004). Internet indirection infrastructure. *IEEE/ACM Transactions on Networking*, 12(2):205–218.

[Stoica et al. 2003] Stoica, I., Morris, R., Liben-Nowell, D., Karger, D., Kaashoek, M., Dabek, F. e Balakrishnan, H. (2003). Chord: A scalable peer-to-peer lookup service for Internet applications. *IEEE/ACM Transactions on Networking*, 11(1):17–32.

[Vogt 2008] Vogt, C. (2008). Six/One Router: A scalable and backwards compatible solution for provider-independent addressing. Em *Workshop on Mobility in the Evolving Internet Architecture (MobiArch)*, páginas 13–18.

[Wetherall 2006] Wetherall, X. Y. D. (2006). Source selectable path diversity via routing deflections. Em *ACM SIGCOMM*, páginas 159–170.

[Xu e Rexford 2006] Xu, W. e Rexford, J. (2006). MIRO: Multi-path Interdomain ROuting. Em *ACM SIGCOMM*, páginas 171–182.

[Yang 2003] Yang, X. (2003). NIRA: a New Internet Routing Architecture. Em *ACM SIGCOMM workshop on Future Directions in Network Architecture (FDNA)*, páginas 301–312.

[Yang et al. 2007] Yang, X., Clark, D. e Berger, A. W. (2007). NIRA: a New Interdomain Routing Architecture. *IEEE/ACM Transactions on Networking*, 15(4):775–788.

Capítulo 3

Novas Arquiteturas de Data Center para *Cloud Computing*

Fábio Luciano Verdi[1], Christian Esteve Rothenberg[2], Rafael Pasquini[2] e Maurício Ferreira Magalhães[2]

[1]Universidade Federal de São Carlos - UFSCar
[2]Faculdade de Engenharia Elétrica e Computação (FEEC) - Unicamp

Abstract

Recently, there has been a change in the way we interact with services and applications. The paradigm (still under evolution) of cloud computing provides services and applications through the Internet intending to offer infinity capacity by using the pay-as-you-go model. The purpose of this work is to present the impacts on the network architectures caused by this new communication model which, on one hand is a good allied of companies such as Google, Microsoft and Yahoo! but, on the other hand, impose technical and market challenges to support a growing demand for cloud services. We will discuss the challenges and limitations of the current network infraestructure, the network requirements for the next generation data centers and the architectures that pursue to attend such requirements.

Resumo

Recentemente tem-se observado uma mudança na forma como interagimos com os serviços e as aplicações. O paradigma (ainda sob evolução) de cloud computing (ou computação em nuvem) fornece serviços e aplicações através da Internet com a promessa de capacidade infinita e modelos de serviço do tipo pay-as-you-go. Este minicurso tem como objetivo apresentar os impactos nas arquiteturas de rede deste novo modelo de comunicação que, por um lado se apresenta como um aliado importante de empresas como Google, Microsoft e Yahoo! mas que, por outro lado, impõe desafios técnicos e de mercado para suportar uma crescente demanda por cloud services. Serão discutidos os desafios e as limitações da infraestrutura tradicional, os requisitos de rede para data centers de nova geração e as arquiteturas de data centers que buscam atender tais requisitos.

3.1. Introdução

Um novo modelo de computação tem surgido e alterado a forma como interagimos com a rede e com os serviços e aplicações. Os *cloud services* [Greenberg 2009] são oferecidos por grandes empresas tais como Google, Yahoo!, Facebook e Microsoft e permitem que aplicações sejam hospedadas e executadas remotamente em um grande *data center*.

O surgimento de serviços populares tais como e-mail baseado na Web, busca (*searching*) e redes sociais, associados ao aumento da conectividade através de banda larga e redes ópticas, impulsionaram o modelo de comunicação centrado no servidor (*server-side*). Cada vez mais o processamento e o armazenamento estão sendo movidos dos PCs para grandes provedores de serviços. Fotos, vídeos e aplicações que antes eram armazenadas e processadas nos PCs, agora migram para serem hospedadas e processadas por provedores sob o formato de serviços Web.

Esta mudança para um modelo centrado no servidor não traz vantagens apenas para o usuário que fica liberado de toda a gerência local necessária para manter as aplicações (intensas configurações e grandes quantidades de *backups*), mas também traz vantagens para os produtores de software [Hoelzle and Barroso 2009]. O desenvolvimento de aplicações é mais simples pois as mudanças e as melhorias nos softwares são feitas em um único local, ao invés de serem feitas em milhões de clientes que possuem diferentes configurações de hardware. Além disso, uma vez que o uso do software hospedado no provedor passa a ser compartilhado por diversos usuários, o custo também se divide entre estes usuários, fazendo com que os valores pela utilização dos serviços sejam reduzidos.

Em termos gerais, a computação se torna mais barata quando vista como um serviço compartilhado. Esta premissa tem sido discutida durante muito tempo mas, até o momento, o modelo centrado no usuário, com máquinas e aplicações individuais, tem sido dominante.

O modelo computacional dos *data centers* começou com os *mainframes* em 1960. Posteriormente, os microcomputadores surgiram no mercado e uma busca constante por altas capacidades de armazenamento se estabeleceu. As estruturas computacionais baseadas nos *mainframes* e nos PCs podem ser vistas como modelos de *data center*s. O primeiro, como um modelo mais concentrado e o segundo, como um modelo distribuído. Esta fase foi seguida pelos sistemas distribuídos baseados no modelo cliente/servidor e, subsequentemente, pelo crescimento explosivo da Internet e da Web. Mais recentemente, a evolução das técnicas de virtualização tem propiciado o desenvolvimento de aplicações que compartilham a mesma infraestrutura de hardware, alavancando o surgimento de soluções para serviços em nuvem.

A diversidade de aplicações que podem ser hospedadas usando o modelo em nuvem inclui desde aplicações comerciais, aplicações de TI e aplicações Web tradicionais até aplicações científicas para processamento paralelo em *batch* e aplicações móveis. Estas diferentes aplicações requerem diferentes arquiteturas de *data centers* e isto tem motivado a pesquisa e o desenvolvimento de soluções que atendam a esta demanda, no sentido de criar mecanismos escaláveis, com alto desempenho e custos mínimos. Isto inclui a pesquisa em infraestrutura voltada para a eficiência energética, cabeamento, resfriamento e, principalmente, infraestrutura de interconexão dos servidores. Neste trabalho enfatizamos os desafios atuais grandes *data centers* em termos de infraestrutura de rede.

Esta seção apresenta as definições, as características essenciais de *cloud computing* e

um panorama geral da área. A Seção 3.2 caracteriza os *data centers* e apresenta a arquitetura atual utilizada nos *data centers* tradicionais destacando as suas limitações e derivando os requisitos de rede para os *data centers* de próxima geração. A Seção 3.3 discute algumas propostas atuais para *data centers* que tentam atender estes requisitos. Finalmente, a Seção 3.4 conclui o capítulo apontando as tendências atuais, os desafios em *cloud computing* e direcionando futuras linhas de pesquisa nesta área.

3.1.1. Definições e terminologias

O que é *cloud computing*?

O termo *cloud computing* (computação em nuvem) possui, como qualquer novo termo[1], várias definições possíveis, muito embora todas relativamente parecidas. O mais importante é entendermos que a definição do conceito está ainda em evolução e novas definições poderão surgir. O trabalho [Vaquero et al. 2009] faz uma análise das definições utilizadas na literatura atual e adota a seguinte opção:

"Cloud computing é um conjunto de recursos virtuais facilmente usáveis e acessíveis tais como hardware, plataformas de desenvolvimento e serviços. Estes recursos podem ser dinamicamente re-configurados para se ajustarem a uma carga variável, permitindo a otimização do uso dos recursos. Este conjunto de recursos é tipicamente explorado através de um modelo pay-per-use com garantias oferecidas pelo provedor através de acordos de nível de serviço (Service Level Agreements-SLAs)."

A Figura 3.1 ilustra o conceito.

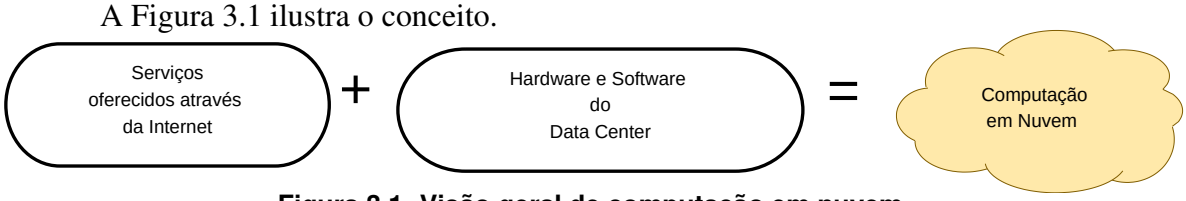


Figura 3.1. Visão geral de computação em nuvem.

Os serviços oferecidos também são conhecidos como *Software as a Service* (o termo será melhor explicado na sequência) e o *data center* engloba todo o hardware utilizado para processar e, também, armazenar os dados associados a eles. Sendo assim, o *data center* e os softwares necessários para gerenciar a oferta de serviços são denominados *cloud*. É importante não confundir os serviços oferecidos pela Internet para os usuários com os softwares e sistemas utilizados nos *data centers* para gerenciamento e fornecimento de tais serviços.

O modelo de computação em nuvem é composto, tipicamente, por cinco características essenciais, três modelos de serviços e quatro modelos de implantação da nuvem [Mell and Grance 2009], conforme descrito abaixo.

- Características Essenciais
  - Serviço sob-demanda: as funcionalidades computacionais são providas automaticamente sem a interação humana com o provedor do serviço;
  - Amplo acesso aos serviços: os recursos computacionais estão disponíveis através da Internet e são acessados via mecanismos padronizados, para que possam ser utilizados por dispositivos móveis e portáteis, computadores, etc.;

[1]Neste trabalho, tanto o termo em inglês quanto o termo em português serão utilizados sem distinção.

  - *Resource pooling*[2]: os recursos computacionais (físicos ou virtuais) do provedor são utilizados para servir a múltiplos usuários, sendo alocados e realocados dinamicamente conforme a demanda do usuário. Neste cenário, o usuário do serviço não tem a noção da localização exata do recurso, mas deve ser capaz de definir a localização em um nível mais alto (país, estado, região);
  - Elasticidade: as funcionalidades computacionais devem ser rápida e elasticamente providas, assim como, rapidamente liberadas. O usuário dos recursos deve ter a impressão de que ele possui recursos ilimitados, que podem ser adquiridos (comprados) em qualquer quantidade e a qualquer momento;
  - Medição dos serviços: os sistemas de gerenciamento utilizados para computação em nuvem controlam e monitoram automaticamente os recursos para cada tipo de serviço (armazenamento, processamento e largura de banda). Este monitoramento do uso dos recursos deve ser transparente para o provedor do serviço, assim como, para o consumidor do serviço utilizado.

- Modelos de serviços

  - Software como um serviço (*Software as a Service* - SaaS): aplicações de interesse para uma grande quantidade de usuários passam a ser hospedadas na nuvem como uma alternativa ao processamento local. As aplicações são oferecidas como serviços pelos provedores e acessadas pelos clientes através de aplicações como o *browser*. Todo controle e gerenciamento da infraestrutura de rede, sistemas operacionais, servidores e armazenamento é feito pelo provedor do serviço. O Google Apps [Google 2010b] é um exemplo de SaaS. A Forrester concluiu que ao usar o Google Apps, as empresas podem economizar de 50% a 70% em comparação com outras soluções de e-mail [Forrester 2010];
  - Plataforma como um Serviço (*Plataform as a Service* - PaaS): é a capacidade oferecida pelo provedor para o usuário desenvolver aplicações que serão executadas e disponibilizadas em nuvem. Neste sentido, surge um outro conceito conhecido como *utility computing*[3], utilizado para denominar toda a plataforma de suporte ao desenvolvimento e fornecimento de aplicações em nuvem. Qualquer aplicação quando desenvolvida precisa seguir um modelo de computação, um modelo de armazenamento e um modelo de comunicação. As plataformas para desenvolvimento de aplicações em nuvem fornecem tais modelos e permitem utilizar conceitos implícitos tais como virtualização e compartilhamento de recursos. As *Utility Computing* mais relevantes atualmente são a AppEngine da Google [Google 2010b] e a plataforma Azure da Microsoft [Microsoft 2010]. Na Seção 3.1.3 apresentaremos as três classes de *Utility Computing* e suas diferenças;
  - Infraestrutura como um Serviço (*Infraestructure as a Service* - IaaS): é a capacidade que o provedor tem de oferecer uma infraestrutura de processamento e armazenamento de forma transparente. Neste cenário, o usuário não tem o cont-

[2]O termo *resource pooling* é utilizado para definir um conjunto de recursos que se comportam como se fossem um único recurso [Wischik et al. 2008]. O objetivo desta técnica é aumentar a confiabilidade, flexibilidade e eficiência do sistema como um todo.

[3]Uma tentativa de tradução para este termo seria computação vista como um utilitário. Entretanto, neste caso, os autores preferem usar o termo em inglês.

role da infraestrutura física mas, através de mecanismos de virtualização, possui controle sobre os sistemas operacionais, armazenamento, aplicações instaladas e, possivelmente, um controle limitado dos recursos de rede. Um exemplo de *Utility Computing* disponibilizada como uma IaaS é a Amazon EC2.

- Modelos de implantação
  - Nuvem privada (*private clouds*): compreende uma infraestrutura de nuvem operada unicamente por uma organização. Os serviços são oferecidos para serem utilizados internamente pela própria organização, não estando disponíveis publicamente para uso geral;
  - Nuvem comunidade (*community cloud*): fornece uma infraestrutura compartilhada por uma comunicade de organizações com interesses em comum;
  - Nuvem pública (*public cloud*): a nuvem é disponibilizada publicamente através do modelo *pay-per-use*. Tipicamente, são oferecidas por companhias que possuem grandes capacidades de armazenamento e processamento;
  - Nuvem hibrida (*hybrid cloud*): a infraestrutura é uma composição de duas ou mais nuvens (privada, comunidade ou pública) que continuam a ser entidades únicas porém, conectadas através de tecnologia proprietária ou padronizada.

Ao analisarmos as definições expostas acima, é possível destacar três novos aspectos em relação ao hardware, introduzidos em *cloud computing*. São eles:

- A ilusão de recurso computacional infinito disponível sob-demanda;
- A eliminação de um comprometimento antecipado por parte do usuário;
- A capacidade de alocar e pagar por recursos usando uma granularidade de horas.

Esta elasticidade para obter e liberar recursos é um dos aspectos chaves da computação em nuvem, sendo uma das principais diferenças quando comparada com computação em grade. A próxima seção faz uma breve comparação entre computação em nuvem, computação em grade e *High Performance Computing* (HPC).

3.1.2. Grades, Computação em Nuvem e HPC

Muitas das características encontradas em grades computacionais também são encontradas na computação em nuvem. Isto ocorre porque ambos os modelos possuem objetivos comuns tais como: redução dos custos computacionais, compartilhamento de recursos e aumento de flexibilidade e confiabilidade. Entretanto, existem algumas diferenças que precisam ser enfatizadas. Estas semelhanças e diferenças têm causado confusão e sobreposição de características e funcionalidades. O trabalho [Vaquero et al. 2009] realiza um estudo das diferentes características associadas com computação em nuvem e as compara com as características associadas com grades computacionais. Apesar do trabalho apontar 12 itens comparativos, abaixo elencamos apenas os mais importantes.

- Modelo de pagamento e origens: as grades computacionais surgiram através de financiamento público, na maioria das vezes patrocinadas por projetos dentro de universidades. O modelo *cloud computing* é motivado por aspectos comerciais onde grandes empresas criam estratégias de mercado com interesses nos lucros. Tipicamente, os serviços em grade são cobrados usando uma taxa fixa por serviço, enquanto que os

usuários dos serviços oferecidos nas *clouds* são cobrados pelo modelo *pay-per-use*. Muitas aplicações não usam a mesma capacidade computacional de armazenamento e recursos de rede. Sendo assim, o modelo de cobrança deve considerar o pagamento separado para cada tipo de recurso utilizado;

- Compartilhamento de recursos: as grades computacionais compartilham os recursos entre as organizações usuárias através do modelo "mais justo possível". A computação em nuvem fornece a quantidade de recursos desejados para cada usuário dando a impressão de recurso dedicado. Esta noção de recurso dedicado é possível através do uso de virtualização, aspecto ainda pouco explorado pelas grades;
- Virtualização: as grades computacionais usam interfaces para esconder a heterogeneidade dos recursos computacionais. A virtualização utilizada em grades computacionais é ainda muito simplista. A virtualização utilizada em *cloud computing* ocorre de forma plena, possibilitando que usuários instalem máquinas virtuais e sistemas operacionais específicos nestas máquinas virtuais. A migração/mobilidade de máquinas virtuais também é um aspecto comum dentro da nuvem e permite a otimização do uso de recursos de energia e resfriamento;
- Escalabilidade e gerenciamento: a escalabilidade em grades ocorre através do aumento no número de nós de processamento. A escalabilidade em *cloud computing* ocorre através de um redimensionamento do hardware virtualizado. O gerenciamento das grades computacionais é dificultado pois não há tipicamente uma única entidade proprietária de todo o sistema. Por outro lado, as *clouds* encontradas atualmente são controladas por uma única entidade administrativa, muito embora exista uma tendência em se criar federações de nuvens;
- Padronização: a maturidade das grades computacionais fez com que vários fóruns fossem criados para a definição de padronização. Neste sentido, esforços para padronização de interfaces para os usuários assim como padronização de interfaces internas alavancaram a interoperabilidade de grades computacionais. Em *cloud computing*, as interfaces de acesso aos serviços são muito parecidas com as interfaces das grades, entretanto, as interfaces internas são proprietárias e dificultam a criação de federação de nuvens. Atualmente há várias iniciativas para definição de padrões para computação em nuvem [Cloud Standards 2010]. Um dos desafios principais é a padronização do formato das imagens virtuais e APIs de migração.

Em termos gerais, grade computacional se refere ao processamento distribuído e paralelo, ou seja, quebrar uma tarefa em várias, distribuir em nós para processamento e então unir as partes para obter o resultado final. Na maioria das vezes, isto significa executar a mesma tarefa em diferentes conjuntos de dados para agilizar o resultado. Para isso, distribui-se a atividade no número máximo de unidades de processamento possível, enquanto que em *cloud computing*, obtém-se a quantidade de recursos suficientemente necessária para a realização da tarefa computacional em um determinado tempo.

Há também distinções entre HPC e computação em nuvem. Tipicamente, HPC se concentra em executar uma única aplicação com alto desempenho, muito similar às grades computacionais. As aplicações HPC podem ser executadas em grade ou utilizar uma nuvem, entretanto, devem seguir as premissas temporais destes modelos. O desenvolvimento de aplicações HPC faz uso de um modelo de programação de baixo nível, enquanto o desenvolvimento de aplicações em nuvem ocorre através da utilização de plataformas com linguagens e

ambientes em alto nível (PaaS) focando na implementação de serviços que serão executados continuamente no *data center*.

Ian Foster em seu blog [Foster 2010] faz uma pequena discussão que compara a execução de uma aplicação em um supercomputador e em uma nuvem. Para os testes, o supercomputador escolhido foi o sistema "Abe" do *National Center for Supercomputing Applications* [NCSA 2010] e a nuvem escolhida foi a Amazon EC2 [Amazon 2010a].

Os resultados mostraram que a execução da aplicação no supercomputador foi extremamente mais rápida. Entretanto, a análise feita por Ian Foster considera o fato de que, na maioria das vezes, a obtenção de recursos computacionais em um supercomputador não ocorre de maneira imediata. Neste sentido, executar uma determinada aplicação em nuvem pode trazer vantagens quando consideramos o tempo total (momento da submissão até o momento da obtenção dos resultados) para concluir a tarefa.

O teste mostrou que a aplicação executada em um supercomputador com 32 processadores precisou de 25 segundos para ser finalizada. Enquanto que a mesma aplicação precisou de 100 segundos na Amazon EC2. O tempo necessário para se obter 32 nós (imagens de máquinas virtuais) no *data center* da Amazon foi de 300 segundos. Sendo assim, o tempo total para concluir a tarefa na Amazon EC2 foi de 100 + 300 = 400 segundos. Por outro lado, a probabilidade de se obter em 400 segundos 32 processadores utilizáveis por 20 segundos no supercomputador mencionado é de 34%, uma taxa relativamente baixa.

Neste sentido, aplicações HPC precisam considerar a possibilidade de utilizarem *cloud computing* ao invés de apostarem unicamente em supercomputadores. Além de obterem um bom tempo de resposta, na maioria das vezes o custo é reduzido.

3.1.3. Classes de computação utilitária (*Utility Computing*)

Atualmente existem três principais plataformas, cada uma disponibilizando níveis diferentes de funcionalidades dependendo do grau de abstração oferecido ao programador: a Amazon EC2 [Amazon 2010a], a AppEngine da Google [Google 2010b] e a plataforma Azure da Microsoft [Microsoft 2010].

Dentre as três plataformas mencionadas anteriormente, a Amazon EC2 é a que oferece a maior liberdade de acesso aos recursos. Uma instância EC2 oferece ao usuário desenvolvedor um conjunto de recursos físicos como se fosse uma máquina real, podendo controlar praticamente todo o software a partir do *kernel*. A API oferecida pela Amazon EC2 é pequena, possuindo poucas chamadas para configuração do hardware virtualizado e não há, em princípio, limite quanto aos tipos de aplicações que podem ser hospedadas pela nuvem da Amazon. Se, por um lado, toda esta flexibilidade oferece ao usuário uma elevada capacidade de desenvolvimento, para a Amazon, tanta flexibilidade dificulta a manutenção da escalabilidade e a gerência de falhas.

Os AWS (*Amazon Web Services*) [Amazon 2010b] oferecem um conjunto de ferramentas e serviços prontos para facilitar a vida dos desenvolvedores de aplicações na nuvem, que podem ou não ser utilizados em conjunto com o EC2. Tais ferramentas e serviços incluem serviços de armazenamento (S3 - *Simple Storage Service*), bases de dados (RDS - *Relational Database Service*, *SimpleDB*), processamento massivo de dados (*Elastic MapReduce*), faturamento (*DevPay*), entre outros.

No outro extremo, encontra-se a AppEngine da Google que oferece uma plataforma para desenvolvimento de aplicações específicas para determinados domínios. A AppEngine é voltada para o desenvolvimento de aplicações Web tradicionais, forçando que tais aplicações sejam estruturadas em duas camadas: camada sem estado e a camada com estado. As aplicações desenvolvidas utilizando a AppEngine devem usar o modelo *request-reply* e, para isso, é de extrema importância levar em conta a quantidade de CPU utilizada pelas requisições realizadas. A AppEngine possui um mecanismo bastante escalável para armazenamento de dados. As aplicações desenvolvidas pela AppEngine usam a MegaStore, uma solução proprietária da Google para armazenamento de dados baseada na BigTable [Google 2010a][4]. Atualmente, as aplicações podem ser desenvolvidas utilizando as linguagens Java e Python. A linguagem Java se integra ao Google Web Toolkit [Google 2010c] e traz um *plug-in* para o Eclipse que permite o desenvolvimento completo de aplicações AJAX.

A plataforma Azure da Microsoft é um ponto intermediário entre a flexibilidade total da EC2 e a especificidade da AppEngine. As aplicações desenvolvidas na plataforma Azure utilizam as bibliotecas .NET e são compiladas para uma linguagem independente de ambiente denominada de *Common Language Runtime*. A plataforma Azure suporta o desenvolvimento de aplicações de objetivo geral tal como a EC2, ao invés de uma única categoria de aplicações como no caso da AppEngine. Permite ainda um certo grau de escolha nas linguagens (PHP, .NET, proprietária) mas não permite o controle do sistema operacional. As bibliotecas fornecem a possibilidade de configuração automática da rede e gerência de falhas, mas requerem que o desenvolvedor defina isto dentro de suas aplicações. A Figura 3.2 apresenta algumas *utility computing* e o modelo de camadas organizado em Infraestrutura (IaaS), Plataforma (PaaS) e Aplicação (AaaS).

3.1.4. Benefícios e oportunidades de novas aplicações

Embora seja possível imaginar a quantidade de aplicações que podem ser desenvolvidas utilizando as plataformas descritas acima e vislumbrar um universo de novos serviços, espera-se que esta diversidade de aplicações cresça muito mais. Abaixo listamos algumas aplicações que podem se beneficiar com *cloud computing* [Armbrust et al. 2009].

- Aplicações móveis interativas: estes serviços tendem a utilizar o modelo em nuvem, pois necessitam estar sempre disponíveis e requerem armazenamento de grandes quantidades de dados. Sensores e, mais recentemente, o conceito de *Internet of Things*, farão com que a produção de dados aumente e necessite ser armazenada em *data centers* geograficamente próximos ao local onde serão utilizados;
- Processamento paralelo em *batch*: muito embora as aplicações tipicamente utilizadas em nuvem sejam interativas, o processamento em *batch* de grandes quantidades de dados é um forte candidato para ser realizado nos *data centers*;
- Computação analítica de dados: um caso especial do processamento em *batch* é a análise de dados de negócios. Neste sentido, uma grande tendência, atualmente, nas empresas é entender o comportamento e o perfil de seus clientes a fim de oferecer novos produtos e serviços. Este tipo de análise permite fundamentar as tomadas de decisões na empresa e organizar estratégias de mercado;

[4]A BigTable é um sistema distribuído de armazenamento desenvolvido pela Google capaz de armazenar grandes quantidades de dados. Atualmente, a BigTable é responsável por armazenar dados de mais de 60 produtos da Google, incluindo Google Analytics, Google Finance, Orkut e Google Earth.

- Aplicações que requerem computação intensiva: vários pacotes matemáticos como Matlab e Mathematica requerem uma capacidade de processamento bastante intensa. Renderização em 3D também exige alta capacidade de processamento. Tais aplicações podem fazer uso do modelo em nuvem para realizarem cálculos complexos, evitando atualizações constantes de hardware em cada *desktop*.

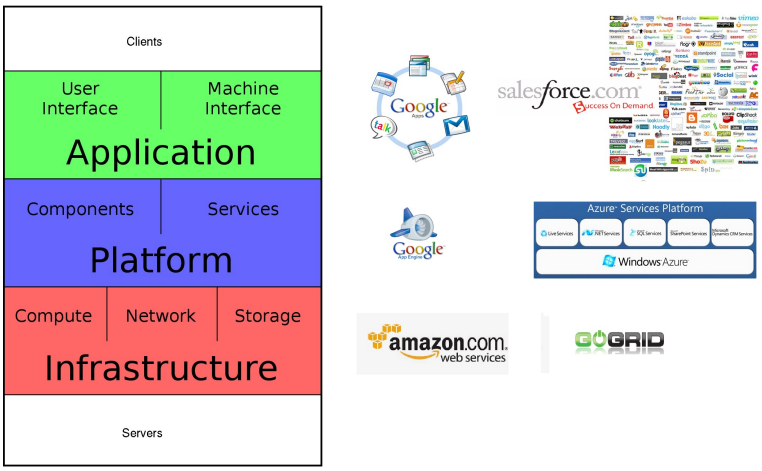


Figura 3.2. Modelo em camadas da computação em nuvem com exemplos de serviços oferecidos. Extraída de [Johnston 2009].

3.1.5. Distribuição de custos

A busca por redução de custos é um dos itens dominantes no projeto das infraestruturas dos serviços em nuvem. Muitos esforços são dedicados na obtenção de soluções mais eficientes, desde a modularização da infraestrutura do *data center*, até componentes *green* que atenuem o consumo da energia em função da carga [Barroso and Hölzle 2007], passando por propostas de encaminhamento de pacotes para aqueles *data center*s onde a energia é mais barata [Qureshi et al. 2009]. A quantificação dos custos associados a um *data center* é uma tarefa complexa que tem sido objeto de recentes estudos.

Segundo a análise de Greenberg [Greenberg et al. 2009a] e os argumentos de James Hamilton [Hamilton 2008], os principais custos de um *data center* da ordem de 50.000 servidores construído seguindo as melhores práticas da indústria (qualidade, alta disponibilidade, etc.), podem ser estruturados conforme a Figura 3.3. Os custos são amortizados assumindo um tempo de vida razoável para os equipamentos adquiridos e a infraestrutura instalada, assim como, um custo de 5% para o dinheiro. Desta forma, pode-se obter uma métrica de custo comum que pode ser aplicada na fase inicial do projeto (por exemplo, aquisição de equipamentos) e durante a operação do *data center* (por exemplo, energia, manutenção, etc.).

Ao observarmos o gráfico acima, constatamos que 62% do custo pertence à infraestrutura de TI (44% em servidores e 18% em equipamentos de rede). Os números específicos podem ser discutidos e variar de um caso para outro. Porém, levando em conta a tendência atual relativa ao aumento dos custos com energia e infraestrutura, enquanto o custo de servidores (medido em trabalho realizado por dólar investido) continua a cair, a conclusão é que os custos totais associados à energia (soma dos custos com a energia consumida, para efetuar a refrigeração e a infraestrutura necessária para distribuição de energia) são hoje comparáveis aos custos dos equipamentos de TI e poderão dominar os custos de um *data center*.

Embora a infraestrutura de comunicação não represente o maior custo, cabe destacar a importância da inovação nas arquiteturas de redes e nos sistemas distribuídos de gerência para a redução dos custos e a obtenção do máximo retorno para cada dólar investido. Uma maneira de se obter isso é oferecendo agilidade aos mecanismos de rede, para que qualquer máquina virtual possa ser instanciada em qualquer servidor físico disponível, independentemente da sua localização na rede, conforme discutiremos mais a frente neste minicurso (veja Seção 3.2.3). Outras propostas de novas arquiteturas de rede para *data center*s [Brandon 2009] sugerem ligar/desligar *switches* que, em função do tráfego, formam caminhos redundantes, apontando para reduções no consumo de energia dos equipamentos de rede na ordem de 50%.

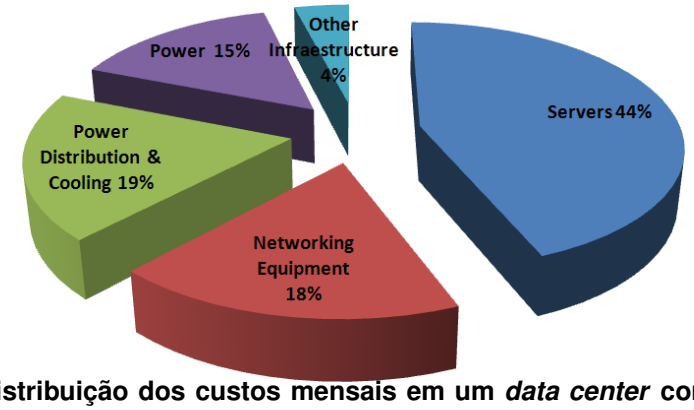


Figura 3.3. Distribuição dos custos mensais em um *data center* com 50 mil servidores. Extraída de [Hamilton 2009a].

3.1.5.1. Eficiência energética

A energia e a refrigeração são dois dos maiores problemas (fontes de despesa) que as organizações de TI enfrentam, de maneira que esses custos devem ser controlados para permitir a expansão e proliferação de mais e maiores *data centers*. *Data centers* que conseguem uma economia eficiente da energia podem gerenciar melhor o aumento de processamento computacional, da rede e as demandas de armazenamento. A redução dos custos de energia se traduz em um menor custo total de infraestrutura, mantendo a oferta de serviços competitiva e capaz de atender às necessidades de futuros negócios.

O Consórcio Green Grid [Christian Belady (ed) 2007] tem reconhecido a importância do estabelecimento de métricas para a eficiência do *data center*, com o objetivo de orientar sobre as tecnologias capazes de melhorar o desempenho por Watt. Idealmente, essas métricas ajudam a determinar se os *data centers* existentes podem ser otimizados antes da ampliação e construção de novas infraestruturas. O Green Grid define duas métricas relacionadas (ver fórmulas abaixo): (1) *Power Usage Effectiveness* (PUE) e (2) a eficiência da infraestrutura do *data center* (*Datacenter infrastructure Efficiency*-DCiE).

PUE = Total Facility Power/IT Equipment Power (1)

DCiE = 1/PUE = IT Equipment Power/Total Facility Power x 100% (2)

Note que nas equações 1 e 2, o "*Total Facility Power*" é definido como a energia dedicada exclusivamente ao *data center*. O "*IT Equipment Power*" é definido como a energia consumida por todo o equipamento que é usado para gerenciar, processar, armazenar, ou encaminhar os dados dentro do *data center*.

O valor PUE ideal é de 1.0, correspondendo a um *data center* onde toda a energia fornecida pela rede elétrica é dedicada para equipamentos de TI, não havendo gasto de energia com refrigeração e nem com distribuição de energia. Um PUE < 1 seria possível com a geração local de energia com base em fontes térmicas residuais, mas atualmente este modelo é comercialmente impraticável de se implementar.

Embora o PUE e o DCiE sejam essencialmente equivalentes, eles podem ser usados para ilustrar a alocação de energia no *data center* de forma diferente. Por exemplo, se um PUE está determinado a ser 3.0, isso indica que a demanda do *data center* é três vezes maior do que a energia necessária para alimentar o equipamento de TI. Além disso, a relação pode ser usada como um multiplicador para o cálculo do impacto real das demandas de energia. Por exemplo, se um servidor exige 500 watts e o PUE para o *data center* é 3.0, então a energia que deverá ser disponibilizada para entregar 500 watts para o servidor é de 1500 watts. Reciprocamente, um valor DCiE de 33% (equivalente a um PUE de 3.0) sugere que os equipamentos de TI consomem 33% da energia no *data center*.

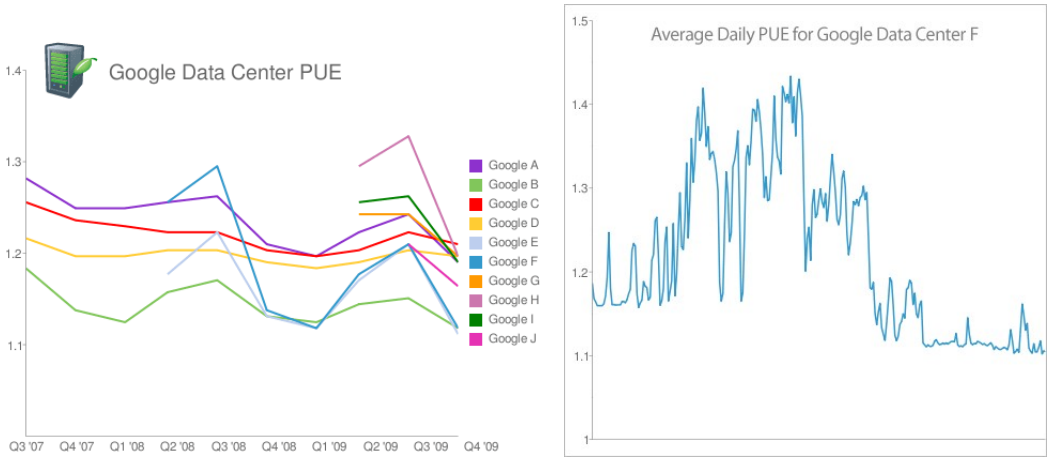
Infelizmente, o PUE médio dos *data centers* convencionais é vergonhosamente elevado. Segundo estudos de 2006 [Lim et al. 2008], 85% dos *data center*s atuais possuem um PUE estimado superior a 3.0, isto é, os sistemas mecânicos e elétricos do *data center* consomem o dobro de energia quando comparados ao equipamento de TI. Estudos mais otimistas mostram um PUE médio de aproximadamente 2.0 para um conjunto de 22 *data centers* pesquisados [Greenberg et al. 2006].

O especialista James Hamilton discute no seu blog [Hamilton 2010a], de forma regular, desafios e tendências no mundo dos *data center*s, incluindo discussões sobre utilização do PUE e estratégias para melhorar a eficiência energética [Hamilton 2009b]. Na realidade, o PUE é um valor dinâmico, que muda com as alterações de carga de trabalho no *data center*. A Figura 3.4(a) apresenta os valores médios de PUE de 10 *data center*s de grande escala da Google. Já a Figura 3.4(b) ilustra uma amostra da variação diária do PUE em um desses *data center*s. O melhor PUE é atingido quando todos os servidores no *data center* estão funcionando perto da capacidade máxima. Sendo assim, desligar os servidores com o intuito de economizar energia acaba, na verdade, aumentando o PUE.

3.1.6. *Comoditização* e consolidação das tecnologias

A infraestrutura dos *data centers* é composta por servidores e redes de conexão, sendo que o custo de manutenção e reposição desta infraestrutura tem se tornado um grande desafio para empresas que oferecem serviços em nuvem considerando que os *data centers* possuem na ordem de 100 mil servidores[5]. Para reduzir tais custos, estas empresas têm se beneficiado do barateamento de equipamentos hoje disponíveis no mercado. Em um primeiro momento, houve uma substituição de servidores de alto desempenho e alto custo por servidores de pequeno porte, porém com baixo custo. Em um segundo momento, a infraestrutura de rede seguiu esta tendência, realizando a substituição dos equipamento de rede. Neste sentido, os provedores de serviços em nuvem estão fazendo uso de um conceito conhecido como *scale-out* ao invés de *scale-up*. *Scale-out* significa instalar equipamentos baratos ao invés de substituir os equipamentos em uso por equipamentos cada vez mais caros. Além disto, o modelo de *scale-out* se beneficia das economias de escala na aquisição massiva de equipamentos, assim

[5]Já em 2006 o Google possuía 450 mil servidores distribuídos em 30 *data centers* [Guo et al. 2008].



(a) Valores médios de PUE de 10 *data center*s. Extraída de [Google 2010d].

(b) Variação diária do PUE do *data center* F. Extraída de [Google 2010d].

Figura 3.4. Valores médios e valores específicos do PUE.

como da consolidação de plataformas por meio das tecnologias de virtualização.

Todo este modelo de utilização de equipamentos menores e mais baratos é conhecido como "comoditização" (*commoditization*). Um dos principais motivadores da *comoditização* dos equipamentos de rede é a disponibilidade de *switches* com preços abaixo de U$$ 100 por porta de 1Gbps e U$$ 1,000 por porta de 10 Gbps [Greenberg et al. 2008]. Apesar destes *switches* não possuírem processamento sofisticado de pacotes e grandes *buffers*, eles oferecem um plano de dados básico operando em altas velocidades. Um outro motivador para utilização deste tipo de equipamento é a capacidade de se criar um plano de controle adaptado às necessidades específicas dos *data centers*, permitindo maior flexibilidade para interconexão dos servidores.

Um outro problema em *data centers* é a capacidade que a infraestrutura de rede oferece para comunicação entre servidores. Estima-se que atualmente os servidores de grandes *data centers* utilizam em média de 5% a 20% de sua capacidade de processamento [Armbrust et al. 2009]. Considera-se um cenário ótimo quando a utilização alcança 30% da capacidade [Greenberg 2009]. Estes números estão consistentes com a realidade uma vez que se tem observado que para muitos serviços o pico de carga excede a média em fatores de 2 a 10, exigindo um super-dimensionamento que suporte tais picos. Esta subutilização da capacidade de processamento dos *data centers* se deve muito a infraestrutura de rede que acaba sendo um ponto de congestionamento para a comunicação entre servidores (mais detalhes na análise de *oversubscription* na Seção 3.2.2.2).

Esta limitação na conectividade entre servidores ocorre devido à infraestrutura de rede hierárquica dos *data centers* atuais, reduzindo a utilização do uso dos recursos computacionais e dificultando o que se denominou de agilidade (*agility*), capacidade de realocar dinamicamente servidores entre os serviços oferecidos pela nuvem, conceito que abordaremos com mais detalhes na Seção 3.2.3.

Neste sentido, a utilização de *switches* "comoditizados" permite que o encaminhamento no plano de dados ocorra somente na camada 2, eliminando a necessidade de

roteadores em vários pontos da rede. Técnicas de engenharia de tráfego tais como *Equal-Cost Multi-Path* (ECMP) são utilizadas permitindo que vários caminhos alternativos entre servidores sejam disponibilizados aumentando assim a capacidade de comunicação e, consequentemente, a capacidade de processamento dos servidores.

3.2. Caracterização dos *data centers* para serviços em nuvem

Com objetivo principal de apresentar os requisitos de rede demandados pelas arquiteturas dos *data centers*, primeiramente apresentamos as características de infraestruturas deste cenário e, então, caracterizamos os serviços e as cargas de trabalho de aplicações típicas. Finalmente, serão discutidas as limitações das abordagens atuais e os requisitos de rede subjacentes em termos de escalabilidade, custo, suporte à virtualização, agilidade, distribuição de carga, tolerância a falhas e segurança.

3.2.1. Infraestrutura dos *data centers*

A infraestrutura da nuvem é formada pelos *data centers* que abrigam os servidores que, mediante diferentes níveis de organização e técnicas de virtualização, oferecem os serviços em nuvem. Portanto, os *data centers* são a manifestação física da computação em nuvem, sendo a infraestrutura de rede a base de comunicações que habilita o paradigma de *cloud computing*, interligando servidores físicos em grande escala. Dependendo do tamanho da própria infraestrutura física e sua localização, os *data centers* podem ser classificados como mega, micro, nano ou baseados em contêineres:

- Mega *data centers*: os chamados mega *data centers* têm na ordem de dezenas de milhares de servidores consumindo dezenas de Mega-Watts de potência. O local para construir estes gigantescos "armazéns" de servidores é escolhido com atenção especial à disponibilidade de recursos como: (1) energia, perto de subestações de grande porte com energia barata e em abundância, ou onde houver fontes de energia naturais, por exemplo, rios ou climatologia adequada no caso de energias renováveis e (2) boa conectividade à Internet. Outros critérios com foco na otimização do custo podem incluir fatores de regulamentação, como incentivos fiscais por emissão reduzida de carbono. Estes mega *data centers* são um ajuste natural para aquelas aplicações de análise de dados (por exemplo, cálculo dos índices de buscadores Web, classificação de documentos ou codificação de arquivos de mídia) ou outras aplicações que requerem enormes quantidades de memória RAM, e/ou exigem um alto número de ciclos de CPU e necessitam de uma grande capacidade de armazenamento em disco rígido. Estes problemas computacionais são atacados de forma distribuída e requerem normalmente uma comunicação intensiva entre os servidores, de modo que o tempo total de computação diminua conforme o atraso entre os servidores é reduzido. Além disso, os custos totais aumentariam se os servidores fossem espalhados por vários centros de dados separados por ligações de longa distância. As aplicações na nuvem são comumente desenvolvidas com base em outros serviços existentes, seguindo uma abordagem de arquiteturas orientadas a serviço (SOA). Um grande número de servidores no mesmo local facilita o projeto dos sistemas e reduz o custo para suportar aplicações com múltiplas dependências e necessidades de comunicação associadas;
- Micro *data centers*: uma área de rápida inovação na indústria é a concepção e implantação de micro *data centers*, com milhares de servidores consumindo centenas de quilowatts. Estes locais são especialmente atrativos para suportar aplicações alta-

mente interativas (por exemplo, e-mail e aplicações de escritório como Google Docs). A localização dos micro *data centers*, também conhecidos como *data centers* satélite, é escolhida estrategicamente em pontos próximos de grandes centros populacionais, para oferecer uma diversidade geográfica que irá minimizar a latência e os custos da rede de trânsito. Hoje, estes micro *data centers* são utilizados principalmente como nós de redes de distribuição de conteúdo (CDNs) localizados em pontos de presença (PoP) de provedores de serviços de Internet (ISP). As principais aplicações são aquelas facilmente distribuídas sem dependências de grandes volumes de trocas entre muitos servidores. Um exemplo típico é o e-mail, onde um fator chave é a interação com o usuário mediante respostas rápidas dos servidores *front-end*. A distribuição geográfica oferece uma opção de projeto com benefícios interessantes em termos de custos, escala, desempenho e confiabilidade. Com os avanços no desenvolvimento e gerência de sistemas de software em nuvem, espera-se que mais *front-ends* de aplicações sejam migrados para estes *data centers*, mantendo os *back-ends* nos centros de maior porte;

- Nano *data centers*: este conceito extremo de *data center* surge da adoção do paradigma *peer-to-peer* (P2P) e da possibilidade de considerar os equipamentos dos usuários finais (por exemplo, os *set-top-boxes*) como uma extensão natural do *data center*, usando os recursos do usuário para migrar a execução de tarefas, armazenar dados e prover serviços através da instanciação de máquinas virtuais. Desta forma, o provedor da infraestrutura da nuvem reduz os custos de adquisição e manutenção dos equipamentos, fazendo com que os serviços e dados fiquem mais perto dos usuários finais. Porém, os desafios de gerenciar um sistema tão distribuído com certas garantias de desempenho, confiabilidade e segurança não são poucos. Isto tem motivado projetos de pesquisa específicos [Nanodatacenters 2010] para este cenário de *cloud computing*, com o apoio de operadoras que já têm este vínculo com o cliente e presença no domicílio do mesmo;

- *Data centers* baseados em contêineres: um outro modelo de disponibilização de *data centers* modularizados é usar contêineres com até 2000 servidores, formando um bloco computacional (ver Figura 3.5) que "só"[6] precisa de energia, água (para refrigeração) e conectividade de rede (fibra óptica) para ser colocado em funcionamento. Esta abordagem modular tem vários atrativos, tanto para os provedores da infraestrutura, quanto para os fornecedores do equipamento, sendo uma forma eficaz de aumentar a capacidade de sua infraestrutura de *data center*. Este tipo de *data center* é uma excelente opção para implantação em locais remotos ou temporários, por exemplo, nas proximidades dos locais de eventos esportivos ou culturais, de forma similar a como estações de telefonia celular são instaladas sob-demanda. Atualmente, a razão mais convincente para sua implantação é a economia de energia (mais do que o aumento de capacidade *ad-hoc*), pois os sistemas de energia são altamente eficientes, podendo economizar de 40% a 50% das despesas operacionais de *data centers* tradicionais. Exemplos comerciais incluem o Portable Modular *data center* da IBM e o Blackbox da Sun. Empresas como Google ou Microsoft já anunciaram o uso deste tipo de infraestrutura.

3.2.2. Visão geral da arquitetura

Independentemente do fator de escala, a infraestrutura do *data center* é constituída por milhares de servidores com suas correspondentes redes de comunicação, subsistemas de ar-

[6] Assumindo os sistemas de software apropriados.

mazenamento, distribuição de energia e extensos sistemas de refrigeração. Há ainda o foco na eficiência dos custos, o fator número um nas decisões de projeto, implementação e procura de novas abordagens na construção e operação desses sistemas. No final das contas, um *data center* é uma fábrica que transforma e armazena bits, tentando maximizar o trabalho útil para cada dólar investido.

Figura 3.5. Um modelo de *data center* baseado em contêineres. Extraída de [Enterprise Control Systems 2010].

3.2.2.1. Infraestrutura de rede

Os servidores físicos são tipicamente montados em conjunto dentro de um *rack* e interligados através de um *switch* Ethernet de acesso. Este *switch* é denominado *Top-of-Rack* (ToR) e usa interfaces de *uplink* de 1 ou 10 Gbps para se interligar com um ou mais *switches* IP/Ethernet de agregação (AGGR), também conhecidos como *switches End-of-Row* (EOR), que agregam os *clusters* de servidores. Este segundo nível de domínio de comutação pode, potencialmente, abranger mais de dez mil servidores individuais. Finalmente, um terceiro nível de *switches* IP/Ethernet é categorizado como *switches*/roteadores de *Core* (CORE), conforme mostra a topologia em camadas da Figura 3.6. A abordagem de topologia em camadas é um fundamento básico dos *data centers* para prover escalabilidade, alto desempenho, flexibilidade, resiliência e facilidade de manutenção. Porém, como veremos nesta seção, a abordagem tradicional (topologia, protocolos de roteamento, separação por VLANs), que tem funcionado razoavelmente bem para os requisitos de *data centers* de redes corporativas, não é suficiente para atender os requisitos dos *cloud data centers* em termos de escala, custo, agilidade e desempenho.

A Figura 3.6 que serve como referência de projetos para *data centers* [Cisco 2007], faz uso de uma sub-camada de serviços representados por um conjunto de ícones e accessíveis pelos *switches* de agregação. Estes serviços são módulos integrados nos *switches* de agregação ou disponibilizados em equipamentos dedicados (*middleboxes* ou serviços externos na Figura 3.6), que proveem serviços tais como balanceadores de carga, roteadores de conteúdo, aceleradores de aplicações (*SSL offload*), segurança (*firewall*, detecção de intrusão, proteção contra ataques de DDoS), análise e monitoramento da rede, etc.

Não estão contemplados na Figura 3.6 os denominados *switches* virtuais rodando nos servidores finais. Em ambientes virtualizados, cada servidor físico pode conter múltiplas máquinas virtuais (VM) gerenciadas por um software de virtualização (*hypervisor*). Este software introduz um *switch* virtual para interligar máquinas virtuais dentro de um mesmo servidor físico e pode estender suas configurações para criar domínios virtuais no nível de *rack*. Esta nova geração de *switches* (por exemplo, Cisco v1000, Open vSwitch) em software é desenvolvida com funcionalidades e interfaces comparáveis aos *switches* físicos, mas com a flexibilidade e velocidade de desenvolvimento de uma implementação em software. Isto facilita extremamente a gerência de redes em ambientes virtuais, além de oferecer numerosas

opções de configuração, isolamento e funcionalidades que seriam intratáveis com técnicas tradicionais de gerência dos *switches* em hardware.

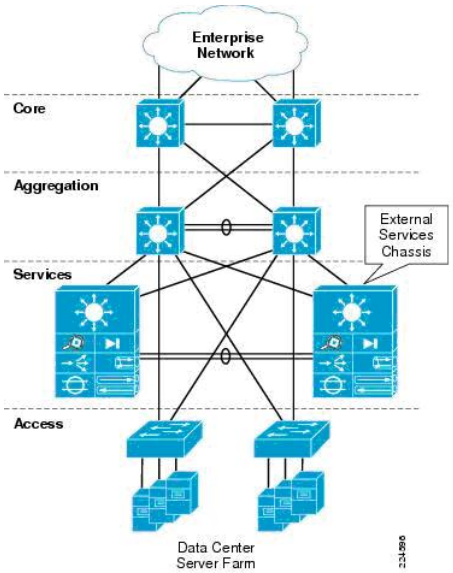


Figura 3.6. Topologia em camadas de um *data center*. Extraída de [Cisco 2007].

Como em outras partes da arquitetura, a escolha da configuração topológica e funcional da camada de rede envolve um compromisso entre capacidade, escala e custo. As propostas de novas topologias de *switches* e mecanismos para o encaminhamento de pacotes diferentes das hierarquias tradicionais serão detalhadas na Seção 3.3. As Figuras 3.7 e 3.8 mostram diferentes topologias em árvore com dois e três níveis de hierarquia, respectivamente. Os modelos de 3 níveis são tipicamente utilizados quando é necessário suportar um grande número de servidores, como no caso dos mega *data centers*.

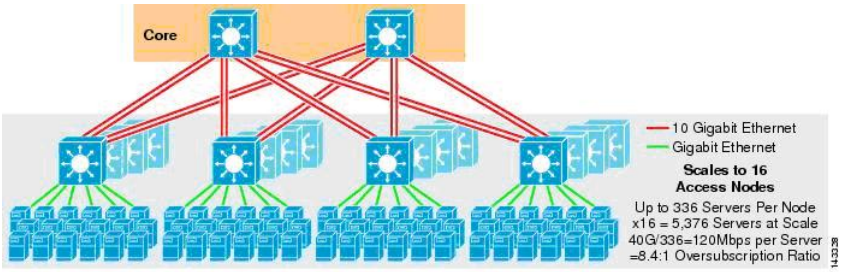


Figura 3.7. Topologia hierárquica de dois níveis e fator de *oversubscription*. Extraída de [Cisco 2007].

Deve ser destacado que os *switches* considerados *commodity*[7] são os que oferecem a melhor relação de custo por porta e por isso são usados como ToRs. *Switches* com um *fan-out*[8] maior, comumente usados na camada de agregação, não mantêm a mesma estrutura de preços. Normalmente, estes *switches* oferecem largura de banda agregada (*bi-section bandwidth*) 10 vezes superiores, mas com um aumento do custo desproporcional, por exemplo, 100 vezes mais caros. Esta descontinuidade nos preços tem motivado a realização de projetos hierárquicos, assim como a utilização exclusiva de *switches commodity* proposto por Al-Fahres [Al-Fares et al. 2008].

[7]Nos dias atuais seria um *switch* Ethernet com até 48 portas de 1-Gbps.

[8]Capacidade que as portas de saída tem de alimentar uma porta de entrada de maior capacidade.

3.2.2.2. Fator de *oversubscription*

Oversubscription no contexto de redes de computadores refere-se à prática de realizar a multiplexação dos recursos de banda para fazer um dimensionamento adequado, que economize a quantidade de enlaces e equipamentos, sem comprometer o desempenho da mesma. Tomemos o seguinte exemplo para ilustrar o fator de *oversubscription* utilizado em *data centers* atuais. Assumindo um *rack* com 40 servidores, cada um interligado a uma porta de um 1Gbps em um *switch* Ethernet de 48-portas de 1Gbps, restarão apenas 8 portas disponíveis para se interligar com a camada de *switches* de agregação. Se todos os servidores quisessem transmitir no máximo da capacidade da interface de rede (40Gbps no total), o tráfego agregado nos *uplinks* seria, no melhor dos casos, de apenas 8 Gbps, o que corresponde a um fator 5 de *oversubscription* para comunicações entre *racks*. Neste tipo de rede, os programadores de aplicações distribuídas devem incorporar estas limitações de recursos da rede na inteligência da aplicação e maximizar o uso de comunicações entre máquinas locais (dentro de um mesmo *rack*), o que complica o desenvolvimento do software e penaliza a eficiência na utilização dos recursos.

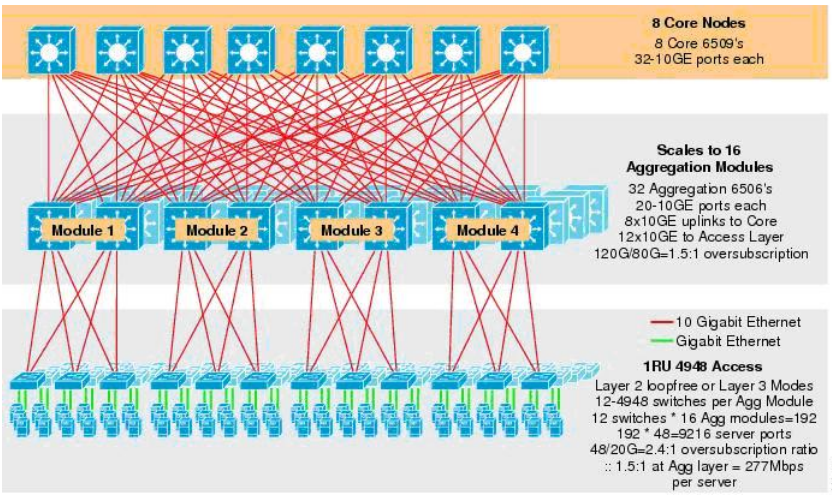


Figura 3.8. Topologia hierárquica de três níveis e fator de *oversubscription*. Extraída de [Cisco 2007].

Aplicando agora o conceito de *oversubscription* ao exemplo da Figura 3.8, o fator total de *oversubscription* será a soma dos valores nos domínios de acesso e de agregação. Neste exemplo, o *switch* ToR é um Catalyst 4948 da Cisco com 48 portas de 1Gbps conectando os servidores e 20 Gbps para *uplink* (cada *switch* possui dois enlaces de 10 Gbps cada), o que corresponde a um fator de *oversubscription* de 2.4:1 (48G/20G) e uma largura de banda real nos servidores de aproximadamente 416Mbps (1Gbps/2.4). Os *switches* de agregação correspondem ao modelo Catalyst 6506 com 20 interfaces de 10Gbps em uma configuração com 8 interfaces para interligar *switches Core* e as outras 12 interligando *switches* ToR, o que resulta em um fator de *oversubscription* de 1.5:1 (120G/80G). Somando ambos os fatores obtemos um fator de *oversubscription* de 3.9:1, o que em termos práticos, limita a largura de banda disponível nos servidores a aproximadamente 277 Mbps (416Mbps/1.5).

Independentemente da capacidade das interfaces de rede dos servidores, por exemplo com a chegada das interfaces de 10Gbps, ao se ter um fator de *oversubscription* superior a 1, a rede se torna um gargalo para matrizes de tráfego onde uma grande parte dos servidores transmitem no máximo da sua capacidade, causando congestionamento nos *switches* e o consequente descarte de pacotes. Arquiteturas de interconexão sem *oversubscription* são bem

conhecidas nas redes de telefonia e nas arquiteturas de HPC, que têm usado topologias de tipo *Clos* ou *fat tree* que garantem a probabilidade de bloqueio zero para qualquer tráfego de entrada [Leighton , Yuan et al. 2007]. Conforme discutiremos nas propostas de novas arquiteturas na Seção 3.3, há interesse em uma rede sem *oversubscription*, porém, os custos associados aos *switches* e cabeamento devem ser justificados pelas demandas de tráfego.

3.2.3. Endereçamento e roteamento IP

Tipicamente, cada aplicação é hospedada em um conjunto específico de servidores (muitas vezes um conjunto de máquinas virtuais). Um *data center* suporta dois tipos de tráfego: (1) o tráfego que entra e sai do *data center* e (2) o tráfego que é gerado e flui apenas internamente ao *data center*. Normalmente, os dois tipos de tráfego são gerados pelas aplicações com algumas especificidades dependendo da aplicação. Por exemplo, em aplicações de busca (*searching*) o tráfego interno domina devido à necessidade de realizar indexações e sincronizações. Por outro lado, em aplicações de vídeo sob-demanda, o tráfego externo prevalece.

Para receber as requisições externas da Internet, uma determinada aplicação possui um ou mais endereços IPs visíveis e válidos publicamente, a fim de que os clientes enviem suas requisições e recebam as respostas. Estes endereços são aqueles obtidos pela resolução de um nome para um endereço IP realizada pelo DNS. Dentro do *data center*, as requisições que chegam da Internet são distribuídas para um conjunto de servidores responsáveis pelo processamento. Normalmente, as comunicações com os usuários externos são atendidas por servidores denominados de *Front-Ends*[9]. Para completar o serviço, estes servidores Web *Front-Ends* tipicamente se comunicam com outros servidores denominados *Back-Ends* para disparar novos (sub-)serviços e acessar os meios de armazenamento distribuídos.

A distribuição das requisições externas aos servidores *Front-Ends* ocorre através do uso de um hardware especializado conhecido como Balanceador de Carga (LB - *Load Balancer*). Neste sentido, uma nova terminologia surge das necessidades de balanceamento de carga: os endereços IPs virtuais (VIPs - *Virtual IPs*) e os endereços IPs diretos (DIPs - *Direct IPs*). Os VIPs representam os endereços públicos, externamente acessíveis na Internet e os DIPs são os endereços internos dos servidores que irão receber e tratar as requisições. A Figura 3.9 ilustra a organização típica de um *data center*.

As requisições de camada 3 (IP) que chegam da Internet utilizando os VIPs são roteadas através dos roteadores de borda e roteadores de acesso até um domínio de camada 2, conforme ditado pelos protocolos de roteamento intra-domínio (tipicamente, IGRP, OSPF). Tais requisições chegam até os LBs que possuem uma lista de endereços DIPs internos que serão utilizados para distribuir cada requisição VIP. Como mostrado na Figura 3.9, todos os servidores que estão conectados a um par de roteadores de acesso pertencem a um único domínio de camada 2. Como há necessidade de rápida convergência durante a ocorrência de falhas e devido às limitações dos protocolos atuais, um único domínio de camada 2 limita-se a possuir no máximo 4000 servidores. Além disso, limitações do protocolo ARP que gera *broadcast* para resolução de endereços, faz com que, tipicamente, o tamanho de uma sub-rede IP não passe de algumas centenas de servidores. Sendo assim, o domínio de camada 2 é dividido em várias VLANs de camada 2, sendo uma sub-rede IP por VLAN, o que fragmenta

[9]Note que, em função do provedor e as características do serviço, os *Front-Ends* podem também estar localizados em micro *data center* satélites para reduzir o tempo de latência com o usuário final.

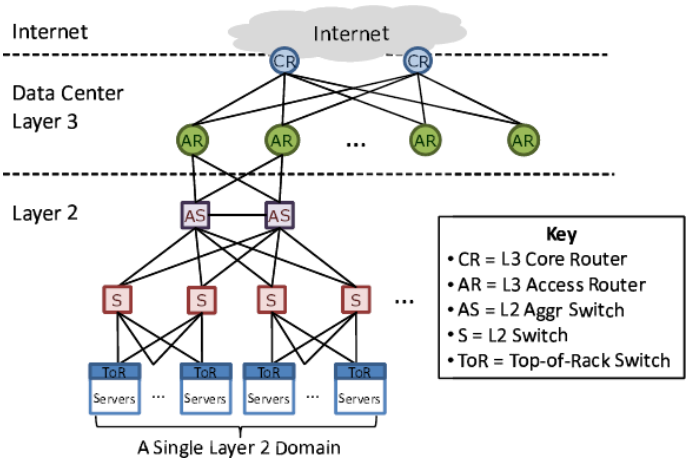


Figura 3.9. Organização de um *data center* tradicional. Extraída de [Greenberg et al. 2008].

os recursos e a capacidade de alocar livremente qualquer endereço IP para qualquer servidor disponível. Neste sentido, surge o termo agilidade definido abaixo:

Agilidade: *é a capacidade de realocar dinamicamente servidores entre os serviços oferecidos pela nuvem com o propósito de otimizar a alocação da carga de trabalho [Greenberg et al. 2009b].*

Sem agilidade, cada serviço deve pré-alocar a quantidade de servidores necessários para atender a demanda em momentos de picos de utilização de um determinado serviço ou em momentos de falhas. Quando um operador de *data center* faz uso da agilidade, é possível atender as flutuações e demandas por serviços individuais através da utilização e distribuição dos serviços entre um conjunto de servidores disponíveis.

Sendo assim, a agilidade em *data centers* tem como objetivo organizar a rede de forma a dar a noção de que cada serviço pode ser alocado em qualquer servidor. Neste sentido, os serviços deveriam utilizar endereços independentes da localização totalmente desacoplados do endereço do servidor. Além disso, se qualquer servidor pode ser alocado para qualquer serviço, é importante que haja isolamento entre os serviços de forma que o desempenho de um serviço não afete o desempenho de outro.

Algumas propostas de novas arquiteturas de *data centers* utilizam este mecanismo. A arquitetura VL2 (apresentada em detalhes na Seção 3.3.2) usa duas famílias de endereços para identificação de *switches* e aplicações. A identificação das aplicações não muda e através de roteamento plano (*flat* ou independente da topologia), a re-alocação de aplicações em diferentes servidores é facilitada.

3.2.4. Software das aplicações em nuvem

A arquitetura típica das aplicações Web é distribuída, seguindo um modelo em camadas: (1) Lógica da aplicação (Ruby on Rails, Scala); (2) caching (Memcache, SQL *query caching*) e (3) *backend* da base de dados (*clusters* RDBMS, CouchDB, BigTable da Google ou Dynamo da Amazon). As interações entre os componentes é realizada através da troca de mensagens assíncronas. Neste ambiente altamente distribuído, cada camada consiste tipicamente de um conjunto de servidores dedicados em executar as respectivas tarefas. A abordagem para au-

mentar a escala da aplicação consiste em aumentar o número de servidores em cada camada, também conhecido como *sharding*.

Esta arquitetura de software funciona bem nas aplicações Web tradicionais, onde as cargas de trabalho, o estado da aplicação e os dados do usuário são facilmente divisíveis. Exemplos deste tipo de aplicação são os serviços de e-mail, onde os usuários acessam os próprios dados, ou dados que são comuns e facilmente armazenáveis em cache, e tipicamente apenas 1-2% dos usuários estão ativos de forma simultânea. Soluções baseadas em DHTs funcionam muito bem para distribuir as cargas de trabalho entre os servidores.

No caso das aplicações de redes sociais, devido ao grande número de usuários, é muito difícil encontrar e gerenciar a quantidade de dependências entre dados e eventos [Pujol et al. 2009]. Por este motivo, este tipo de aplicação é dificilmente escalável, apresentando desafios na distribuição e atualização dos dados entre os servidores, o armazenamento em cache das informações em RAM para serem acessíveis de forma rápida e na propagação dos eventos de atualizações (por exemplo, *Push-on-Change, Pull-on-Demand*). Serviços como Facebook, MySpace, Orkut, Digg ou Twitter são complicados de gerenciar e escalar [Hoff 2009b]. Para termos uma noção da ordem de magnitude, o Facebook possui mais de 30 mil servidores, 300 milhões de usuários ativos e 80 bilhões de fotos. Seus servidores processam 600 mil fotos por segundo, possuem no total 28 TB de cache e produzem 25TB de dados de logs por dia.

Em um site social como o Orkut, considerando um usuário com 200 amigos, cada vez que ele entra na página principal, a aplicação deve coletar simultaneamente o estado dos 200 amigos e apresentar as atualizações. Isto se traduz em 200 requisições simultâneas, mais o processamento das respostas e outra série de serviços que devem ser invocados para obter informações e serviços adicionais. Após o processamento coletivo, o servidor Web *front-end* pode apresentar a página inicial em um tempo razoável.

Usuários com um grande número de seguidores geram bastante processamento nos *data centers* em cada *post*. O grafo da rede social de cada usuário deve ser percorrido para avaliar quem deve receber o *post* (isso ocorre com base nas preferências e configurações do usuário). Atravessar tal grafo em tempo real é uma tarefa que exige processamento. Por exemplo, um usuário da rede social Digg que possui 100 seguidores, gera 300 milhões de "diggs" por dia, 3 mil operações de escrita por segundo, 7GB de dados armazenados por dia e 5TB de dados espalhados entre os 50-60 servidores.

O site social MySpace [Hamilton 2010b] possui 130 milhões de usuários ativos. Aproximadamente 40% da população dos EUA tem conta no MySpace e uma média de 300 mil novos usuários são criados por dia. A infraestrutura é formada por 3 mil servidores Web, 800 servidores de cache e 440 servidores SQL (hospedando mais de 1.000 bancos de dados). Cada servidor SQL roda em um HP ProLiant DL585 (4 processadores *dual core* AMD, 64 GB de RAM). O sistema de armazenamento possui 1.100 discos em uma SAN (*Storage Area Network*), totalizando 1PB de dados dos servidores SQL.

Nessa escala, toda otimização é pouca e já existem trabalhos na área de arquiteturas de aplicações distribuídas otimizadas para redes sociais. Por exemplo, o *One-Hop Replication* (OHR) [Pujol et al. 2009], que considera o particionamento dos dados levando em conta a relação semântica e a importância de cada usuário neste tipo de rede.

Também há um debate intenso sobre novas tendências tecnológicas que não utilizam o modelo tradicional de bases de dados baseadas no modelo SQL. Os atributos das RDBMS com semânticas transacionais do tipo ACID trazem uma rigidez na definição dos esquemas de dados que dificultam a escalabilidade horizontal destes sistemas. Isto tem motivado o surgimento de sistemas "No-SQL", tais como o SimpleDB da Amazon, a BigTable da Google e outros vários projetos No-SQL [No-SQL 2010].

No que se refere a rede, esta deve garantir uma latência baixa e constante para qualquer par de nós que queiram se comunicar, independentemente da sua localização (*intra-rack*, *inter-rack*, *inter-switch*), uma característica que simplifica o projeto e a escalabilidade das aplicações em nuvem.

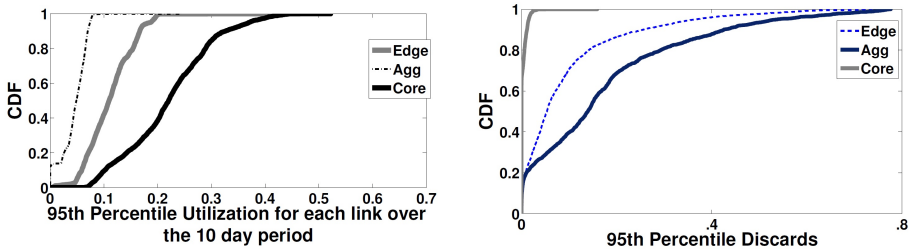
3.2.5. Caracterização do tráfego

Estudos recentes [Benson et al. 2009a, S. Kandula and Patel 2009] têm revelado detalhes sobre as características do tráfego de *data centers* em operação. A conclusão principal é que a matriz de tráfego é altamente variável e dominada por rajadas (picos). Estes fatores dificultam a previsão do comportamento e a aplicação de técnicas tradicionais de engenharia de tráfego, demandando novos mecanismos para reduzir os problemas de congestionamento e prover garantias de uniformidade nas taxas de transferência entre qualquer par de servidores. Além disto, a taxa de utilização dos recursos deve se manter alta e os custos sob controle. Outro dado interessante, referre-se à distribuição do volume, sendo que apenas 20% do tráfego total tem destino/origem na Internet pública. O restante é fruto do desdobramento de novas requisições internas e outros serviços de suporte às operações. Em média, cada servidor contribui com 10 fluxos de tráfego simultâneos.

Com base em dados coletados em 19 *data centers* em funcionamento, o trabalho [Benson et al. 2009a] apresenta um estudo macroscópico das variações temporais e espaciais do tráfego, assim como, as perdas observadas na malha interna dos *switches*. Entre as observações, convém destacar que a carga média nos *switches* do *core* é superior e diminui progressivamente à medida que descemos em direção aos *switches* de acesso (ver Figura 3.10(a)). Por outro lado, as maiores perdas, em média, são superiores nos ToRs e mais reduzidas no *core* (ver Figura 3.10(b)), o que sugere que o tráfego nos enlaces de acesso e agregação possuem um comportamento típico de rajadas. Outra importante observação é que uma pequena fração dos enlaces apresenta perdas muito maiores. Isto indica que seria possível achar rotas por caminhos alternativos que evitem a maior parte das perdas resultantes de *switches* congestionados. Trabalhos recentes que exploram esta abordagem incluem o conceito de *Flyways* de Kandula [Kandula et al. 2009] onde enlaces sem-fio de 60 GHz entre os ToRs são estabelecidos dinamicamente após a detecção de congestionamento. Já explorando um plano ótico reconfigurável, o trabalho [Wang et al. 2009] propõe o estabelecimento dinâmico de caminhos óticos para aliviar os pontos de congestionamento.

Outra observação é que o tráfego nos ToRs segue padrões do tipo ON-OFF, onde os diferentes períodos e o intervalo de tempo entre pacotes podem ser caracterizados como uma distribuição *log-normal*. Os períodos ON-OFF são tipicamente da ordem de milissegundos e, em conjunto, são observados picos de tráfego intensos de duração inferior a 10 segundos. Embora técnicas de balanceamento de carga e re-roteamento possam ser definidas para evitar estes períodos de congestionamento, para serem efetivas, tais técnicas devem ser capazes

de tomar decisões com uma frequência da ordem de segundos. Desta forma, novos mecanismos de engenharia de tráfego para *data centers* devem ser desenvolvidos, para garantir o balanceamento da carga e reagir de forma mais sensível (maior granularidade) que as abordagens tradicionais de redes WAN, uma linha de pesquisa explorada em [Benson et al. 2009b].



(a) Utilização dos enlaces nas diferentes camadas dos *data centers* estudados. Extraída de [Benson et al. 2009a].

(b) Perda de pacotes nas diferentes camadas dos *data centers* avaliados. Extraída de [Benson et al. 2009a].

Figura 3.10. Utilização média e perda de pacotes nas camadas.

Já o estudo apresentado em [S. Kandula and Patel 2009] consiste na instrumentação de 1500 servidores em um *cluster* de *data center* em operação, monitorado durante 2 meses para obter os perfis de tráfego e os padrões e condições de congestionamento. Seguindo uma abordagem baseada na instrumentação dos próprios servidores, ao invés de coletar dados dos *switches*, os autores argumentam que podem ser obtidas métricas mais ricas e confiáveis, evitando a falta de precisão das estatísticas de amostragens por fluxos dos *switches* e o *overhead* de técnicas baseadas em *Deep Packet Inspection* (DPI). A carga de trabalho no *cluster* é fundamentalmente de aplicações de processamento de dados baseadas no modelo MapReduce, onde basicamente os dados a serem processados são divididos em pequenas sub-tarefas, distribuídas entre diferentes servidores escravos, conforme agendado por um servidor mestre. Finalmente, os resultados são agregados e retornados para a aplicação de origem.

A Figura 3.11 mostra o tráfego em $log_e(Bytes)$ trocado entre pares de servidores durante períodos de 10 segundos. A ordenação dos servidores nos eixos replica a localização física dos servidores nos *racks* (20 servidores/*rack*). O trabalho ressalta a identificação de dois padrões de tráfego: (1) *Work-Seeks-Bandwidth* e (2) *Scatter-Gather*:

- *Work-Seeks-Bandwidth*: as formações quadráticas na diagonal do gráfico correspondem ao tráfego trocado entre servidores dentro do mesmo *rack*, como consequência de projetos de programação voltados para a otimização dos recursos da rede. Sendo assim, a distribuição de cargas de trabalhos que envolvem a troca de um volume alto de dados é preferencialmente realizada nas áreas onde a largura de banda disponível entre servidores é maior (dentro do *rack*). Esta é uma prática comum em topologias do tipo árvore com fatores de *oversubscription*, que reduzem a largura de banda efetiva entre os servidores, forçando as aplicações a alocarem o processamento primeiramente em servidores dentro do mesmo *rack*, depois dentro da mesma VLAN e, de menor preferência, em servidores mais distantes. É por este motivo que os autores chamam este padrão de tráfego de *work-seeks-bandwidth*;
- *Scatter-Gather*: o segundo padrão observável da figura vem representado pelas linhas verticais e horizontais, onde um servidor empurra (ou puxa), dados a vários servidores dentro de todo o *cluster*. Este é um claro indicativo das operações de *map* e *reduce* de software de processamento distribuído, onde os dados correspondentes às requisições

são divididos em pequenos pedaços, cada um dos quais é processado por diferentes servidores e os resultados são posteriormente agregados. Por isso o nome de *scatter-gather* ou (dispersão-recolhimento).

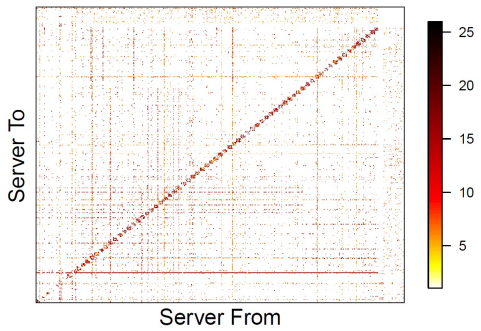


Figura 3.11. Os padrões *Work-Seeks-Bandwidth* e *Scatter-Gather* analisados entre pares de servidores de um *data center*. Extraída de [S. Kandula and Patel 2009].

Desta análise do perfil de tráfego, fica evidente outro requisito das arquiteturas de *data center*: alta largura de banda com taxa de transferência uniforme, independentemente da localização do par de servidores dentro da topologia da rede. Desta forma, (1) qualquer servidor disponível pode ser alocado para as tarefas de computação, liberando a programação do software da complexidade de tratar com a localização dos servidores e (2) o tempo total para conclusão das operações em paralelo não é afetado pelo atraso na obtenção de resultados parciais devido à capacidade da rede.

A Figura 3.12 destaca como o perfil de tráfego agregado varia com uma alta frequência. Os dados da figura correspondem a 165 milhões de fluxos, o total de um dia de operação do *cluster*, revelando que a maior parte dos fluxos são de curta duração (80% duram menos de 10s) e há poucos de longa duração (menos de 0.1% duram mais de 200s). Além disso, mais de 50% dos bytes estão em fluxos que duram menos de 25 segundos.

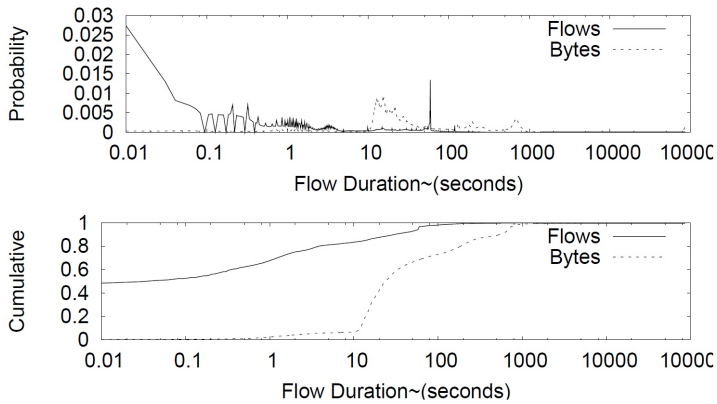


Figura 3.12. Características de tráfego dos *data centers* avaliados. Extraída de [S. Kandula and Patel 2009].

Como já foi observado anteriormente, isto tem implicações interessantes para as técnicas de engenharia de tráfego. Soluções baseadas em decisões centralizadas com visão global da rede são difíceis de implementar, por motivos de escala (quantidade de novos fluxos por segundo) e pela rapidez com que as decisões devem ser tomadas para não atrasar o início dos fluxos. Contribuindo para este desafio, os fluxos de larga duração não são correspondidos com aqueles de maior volume de tráfego. Desta forma, não é suficiente fazer um agendamento especial para este tipo de fluxo "problemático".

A Figura 3.13 ilustra como varia o tráfego no *data center* com o tempo. A sub-figura superior mostra o tráfego agregado de todos os pares de servidores durante um período de 10 horas. Podemos observar que o tráfego muda muito rapidamente, com picos que correspondem a mais da metade da capacidade total de banda da rede. Comunicações *full-duplex* entre os pares de servidores não são a regra, já que tipicamente os papéis de fonte e consumidor dos dados permanecem fixos. Isto indica que durante vários momentos durante um dia típico, os enlaces são utilizados perto da sua capacidade máxima. A outra dimensão na mudança do perfil de tráfego é em relação aos participantes atuais. Embora o tráfego agregado total se mantenha constante, o conjunto de servidores que está se comunicando pode sofrer variações, conforme indica a sub-figura inferior.

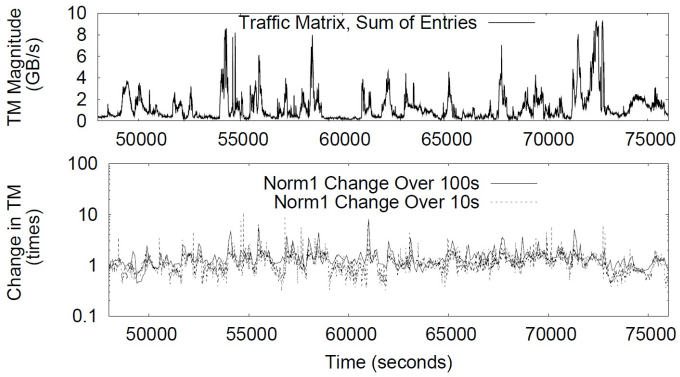


Figura 3.13. Variação do tráfego em termos de quantidade de dados e número de participantes. Extraída de [S. Kandula and Patel 2009].

Em relação aos eventos de congestionamento observados na rede (*hot-spots*), foi avaliada a utilização média dos enlaces com base em uma constante C[10]. A Figura 3.14 ilustra a evidência de numerosos períodos onde a utilização dos enlaces é muito alta (maior do que C). Dentre os 150 *switches* que interligam as 1500 máquinas, 86% dos enlaces sofreram congestionamento de pelo menos 10 segundos e 15% apresentaram períodos de congestionamento de pelo menos 100 segundos. Os períodos curtos de congestionamento (círculos azuis, 10s de alta utilização) estão correlacionados com dezenas de enlaces congestionados, o que indica que são consequência de picos de demanda da aplicação. Por outro lado, períodos largos de congestionamento tendem a se localizar em um número mais reduzido de enlaces.

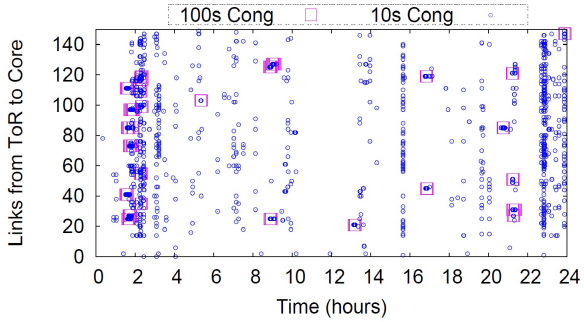


Figura 3.14. Quando e onde o congestionamento ocorre. Extraída de [S. Kandula and Patel 2009].

A Figura 3.15 mostra que a maioria dos períodos de congestionamento tendem a ser de curta duração. De todos os eventos superiores a um segundo, mais de 90% não superam os 2

[10]No artigo foi utilizado um valor de C = 70%, mas a escolha de um limite de 90% ou 95% tem resultados qualitativamente semelhantes.

segundos. Períodos longos de congestionamento também foram observados, contabilizando um total de 665 episódios únicos de congestionamento por mais de 10s. Alguns poucos duraram várias centenas de segundos e o maior durou 382 segundos.

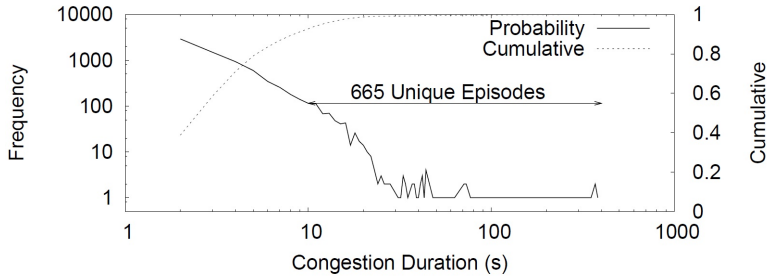


Figura 3.15. Tamanho dos eventos de congestionamento. Extraída de [S. Kandula and Patel 2009].

Idealmente, a rede deve ser operada no maior grau de utilização possível sem prejudicar a taxa de transferência entre os nós. Períodos prolongados de baixa utilização da rede podem significar (1) que a aplicação exige mais de outros recursos, como CPU e disco de rede e está esperando por respostas de algum servidor ou (2) que a aplicação não está otimizada para fazer um melhor (maior) uso dos recursos da rede (banda). Trabalhos recentes [Sonnek and Chandra 2009] têm tentado encaixar VMs com requisitos de CPU, disco (I/O) e rede (I/O) de forma ortogonal, a fim de otimizar a utilização dos recursos no *data center*, uma área de muito interesse e ainda com oportunidades de pesquisa.

3.2.6. Limitações das arquiteturas de rede tradicionais

As práticas atuais de engenharia do *data center* e, em especial, a organização hierárquica L2/L3 da arquitetura de rede, causam uma série de problemas que dificultam a agilidade do sistema (Seção 3.2.3) e apresentam limitações que são resumidas da seguinte forma:

- Fragmentação dos recursos: o mecanismo de roteamento das arquiteturas convencionais atribui, normalmente, endereços IP que possuem significado topológico (aderentes à organização da rede) e são divididos em VLANs, criando um cenário de elevada fragmentação dos servidores. Tal fragmentação introduz rigidez na migração de máquinas virtuais entre VLANs, uma vez que a migração exige a troca do endereço IP para aderir à nova posição topológica. Se uma determinada aplicação cresce e necessita de mais recursos (servidores), tal aplicação não pode utilizar servidores disponíveis e ociosos localizados em outro domínio (sub-rede) de camada 2, causando uma subutilização dos recursos do *data center*;
- Alocação Estática de serviços: as aplicações ficam mapeadas a determinados *switches* e roteadores e necessitam utilizar VLANs para poderem acessar os servidores dedicados a uma aplicação. Apesar das VLANs fornecerem isolamento e segurança, na prática elas fazem com que o tráfego se concentre nos enlaces do topo da hierarquia, causando congestionamento. Além disso, esta alocação estática dificulta a re-alocação dinâmica de serviços. Quando um determinado serviço enfrenta sobrecarga, a intensa utilização da rede faz com que os serviços que compartilham a mesma sub-rede do serviço sobrecarregado também sejam afetados;
- Vazão e latência entre servidores: as arquiteturas convencionais não oferecem capacidade de transmissão suficiente entre os servidores. Organizadas em topologias em

forma de árvores, apresentam taxas de sobrecarga no encaminhamento de tráfego entre diferentes ramos da árvore na ordem de 1:5, ou mais, e concentração de tráfego no mais alto nível da árvore variando entre 1:80 e 1:240 [Greenberg et al. 2009b]. Como consequência, a latência e a vazão se tornam não uniformes dependendo do par de servidores, o que afeta negativamente o desempenho das aplicações nos padrões de tráfego observáveis no *data center*. Além disso, devido à natureza hierárquica da rede, a comunicação entre servidores localizados em diferentes domínios de camada 2 (diferentes VLANs) deve ocorrer através de roteamento em camada 3. Para isso, os roteadores do topo da hierarquia devem possuir alta capacidade de processamento e grandes *buffers*, aumentando ainda mais os custos de TI do *data center*;

- Escalabilidade: os protocolos atuais de roteamento, encaminhamento e gerenciamento não oferecem a escalabilidade necessária para atingir a ordem de grandeza necessária nos *data centers*. As soluções atuais de camada 2 utilizam *broadcast*, criando um cenário não escalável devido ao elevado nível de sinalização. Por outro lado, as soluções de camada 3 exigem a configuração dos equipamentos, seja na definição de sub-redes nas quais os equipamentos estão incluídos ou mesmo na sincronização de servidores DHCP para efetuar a correta atribuição de endereços. Para se ter uma idéia do tamanho destas redes e da complexidade envolvida, considere um *data center* com 100.000 servidores, cada um executando 32 máquinas virtuais. Isto se traduz em mais de três milhões de endereços IP e MAC em um único *data center*;
- Custo dos equipamentos de rede: os balanceadores de carga (LB) tradicionais são equipamentos caros, utilizados em pares em uma configuração 1+1 para tolerância a falhas. Quando estes LBs já não suportam mais a carga, eles são substituídos por novos com mais capacidade, utilizando o modelo *scale-up* ao invés do modelo *scale-out* (mais barato). Este é o modelo seguido também na utilização de *switches* tipo *high-end* de alto desempenho em termos de encaminhamento e "bufferização", com capacidade para suportar picos de processamento e alto consumo de banda;
- Eficiência energética: os *data centers* tradicionais usam mecanismos básicos de refrigeração seguindo a premissa de que se o *data center* cresce, instala-se mais equipamentos de refrigeração, causando um impacto significativo no consumo de energia mensal. Além dos aspectos de refrigeração e sistemas de distribuição da energia na infraestrutura, ainda há uma margem alta de eficiência energética que não é atingida pelos servidores e os equipamentos de rede atuais. Especialmente em relação à proporcionalidade do consumo de energia em função da carga de trabalho pois, nos projetos atuais, o consumo dos equipamentos (e da rede como um todo) trabalhando com uma carga mínima, não consomem proporcionalmente menos energia do que trabalhando ao máximo da sua capacidade.

Conclui-se, baseado nos comentários anteriores, que as técnicas convencionais (*Spanning Tree*, VLAN, criação de sub-redes) não oferecem o desempenho (alta taxa de transferência uniforme fim-a-fim e baixa latência para qualquer par de nós da estrutura) nem a agilidade (flexibilidade na gerência do endereçamento IP/Ethernet) necessários. Vale a pena destacar que estas limitações não são resolvidas simplesmente com um aumento da capacidade de transmissão dos enlaces. Embora uma baixa latência entre servidores seja um dos objetivos, uma latência determinística e consistente entre qualquer par de servidores é igualmente importante para suportar as aplicações em nuvem.

3.2.7. Objetivos e requisitos das arquiteturas de rede para *data centers*

Além dos requisitos óbvios de confiabilidade e desempenho, uma série de novos requisitos em termos de arquitetura e sistema são fundamentais para aumentar a escalabilidade e a eficiência e reduzir os custos dos *cloud data centers*. Um fator chave é habilitar o desacoplamento das aplicações da própria infraestrutura e transformar o *data center* (como um grande computador) em um sistema ágil, de alto desempenho e eficiente. Estes *data centers*, porém, não devem ser confundidos arquiteturalmente com a Internet, ou simplesmente reduzidos a técnicas de virtualização (servidor, armazenamento, rede) e segurança avançadas.

As diferenças fundamentais em relação à infraestrutura de rede em um *data center* comparadas com redes tradicionais locais ou intra-domínio são: (1) a topologia é definida pelo arquiteto da rede; (2) o número de nós pode ser alto, mas é conhecido; (3) serviços de *broadcast/multicast* não são requeridos; (4) os aspectos de segurança ficam simplificados (não há ataques maliciosos, apenas pequenos erros de configuração) e (5) há opções de controle sobre a distribuição das aplicações para minimizar a possibilidade de gargalos.

As redes para *data centers* de grande escala, como as utilizadas em infraestruturas para computação em nuvem, possuem uma contribuição importante para o custo total da infraestrutura. Especialmente os *switches*/roteadores de grande porte têm custos por porta muito altos comparativamente. O software dos roteadores tem evoluído de forma fechada e proprietária, gerando um código legado desnecessariamente grande e complexo sendo apontado como a causa comum de muitos dos problemas de confiabilidade. O serviço de manutenção, assim como, as requisições de personalização e novas funcionalidades são providas apenas pelo fornecedor do equipamento e sempre requerem que o equipamento seja enviado até o local de atendimento.

Estes fatores associados às limitações da arquitetura de rede tradicionais, têm motivado a procura por novos projetos que atendam melhor os requisitos, aumentem o desempenho, melhorem a confiabilidade e reduzam custos. Desta forma, os novos projetos podem ser feitos na medida em que as necessidades particulares do provedor da infraestrutura aumentem e novos serviços sejam ofertados. Isto evita que os provedores paguem por funcionalidades dos equipamentos que não são necessárias e que ficam ociosas na maior parte do tempo. Desta forma, podemos definir algumas metas de projeto das arquiteturas de rede para *cloud data centers*:

- Redução da dependência de *switches* de grande porte no *core* da rede;
- Simplificação do software de rede (plano de controle, pilha de protocolos);
- Aumento da confiabilidade do sistema como um todo;
- Evitar que a rede seja o gargalo do sistema e simplificar o trabalho do desenvolvimento de aplicações;
- Redução dos custos de capital e operacionais.

Com base nas limitações e metas apontadas acima, compilamos os requisitos da arquitetura de rede e os dividimos da seguinte forma:

- Agilidade: endereçamento e encaminhamento de pacotes que permitam levantar (e mover) qualquer máquina virtual em qualquer servidor físico;

- Escalabilidade: roteamento e endereçamento escaláveis, tabelas de encaminhamento (L2 Ethernet) com tamanho gerenciável e tempos de convergência aceitáveis. O objetivo final é poder escalar a uma ordem de centenas de milhares de servidores e milhões de máquinas virtuais. Sendo assim, dois aspectos precisam ser considerados: o tamanho das tabelas de encaminhamento e os *broadcasts*, dois grandes problemas para redes de grande escala;
- Desempenho: vazão e latência uniformes e constantes para qualquer padrão de tráfego (1:1, 1:M, N:N) e entre qualquer par de servidores. Garantir o isolamento entre tráfegos e evitar pontos de congestionamentos. A capacidade de comunicação entre servidores deve ser limitada apenas pela capacidade das interfaces;
- Controle: flexibilidade para inserção de serviços de *middleboxes*, monitoramento e resolução de problemas. A infraestrutura também deve ser capaz de incorporar novos servidores sem a necessidade de considerar a topologia. O controle dos aspectos de segurança deve estar incorporado aos sistemas de gerência;
- Confiabilidade: o projeto da infraestrutura deve suportar falhas de servidores, *switches*, etc. Devido ao grande número de equipamentos, as chances de falhas são maiores e precisam ser contornadas para que o sistema se mantenha operacional;
- Custo: baixo custo de capital e operacional. Melhorar os esforços de configuração através de mecanismos automatizados ajudam a reduzir o custo de manutenção dos sistemas. Usar hardware *comoditizado* reduz o custo de capital. Isto envolve também novos métodos de estruturação e cabeamento, procurando modularidade da infraestrutura do *data center*, que contribuam para uma redução dos custos. Além disso, melhorar a eficiência energética é fazer com que, idealmente, o consumo de energia seja proporcional à carga de trabalho, um problema bastante desafiador.

Cada um destes requisitos pode ser resolvido separadamente com as tecnologias de rede atuais. Porém, satisfazer todos eles ao mesmo tempo é uma tarefa de grandes desafios, principalmente porque atender a um determinado requisito implica, muitas vezes, em não conseguir atender a outro. Por exemplo, um fator de *oversubscription* alto resulta em uma redução do desempenho real entre os servidores em relação a algumas demandas de tráfego. Ao mesmo tempo, um projeto de rede sem *oversubscritpion* implica em um alto custo devido à quantidade necessária de elementos de rede.

A próxima seção apresenta algumas das principais arquiteturas de *data centers* atuais que buscam atender aos requisitos de rede listados acima. Algumas arquiteturas privilegiam certos requisitos em detrimento de outros. Entretanto, como veremos, a maioria das arquiteturas atende em maior ou menor grau a todos os requisitos citados.

3.3. Novas propostas de arquiteturas para *data centers*

Atualmente existe um grande esforço da comunidade (indústria e acadêmica) em busca do desenvolvimento de novas arquiteturas de *data centers* voltadas para a resolução dos problemas citados nas seções anteriores. Nesse sentido, apresentamos quatro arquiteturas que também podem ser consideradas como as mais significativas e que representam o universo das pesquisas realizadas nesta área. Estas quatro propostas envolvem as principais características necessárias para suportar os *cloud data centers* e servem como base para compreendermos o que de fato é necessário neste tipo de infraestrutura.

3.3.1. Monsoon

Conforme discutido na Seção 3.2, as aplicações contidas nos *data centers* atuais sofrem com a fragmentação interna dos recursos, rigidez e limitação de banda que são impostos pela arquitetura de rede que conecta os servidores. Por exemplo, arquiteturas convencionais de rede utilizam mecanismos estáticos para mapear os serviços oferecidos pelo *data center* em VLANs, limitadas em algumas centenas de servidores devido ao nível de sinalização (*overhead*) gerado no plano de controle da rede. Além disso, a utilização de equipamentos no nível IP para efetuar a distribuição de tráfego entre diversas VLANs, em conjunto com os balanceadores de carga necessários para espalhar as requisições entre os servidores, elevam o custo de implantação dos *data centers*. Basicamente, este cenário é caracterizado pela concentração de tráfego em alguns poucos equipamentos, resultando em frequentes substituições de hardware para atender a nova demanda do *data center*.

A filosofia defendida pelo Monsoon [Greenberg et al. 2008] é a *comoditização* da infraestrutura como forma de obter escalabilidade e baixo custo, ou seja, o Monsoon adiciona equipamentos novos e baratos (*scale-out*) para atender a nova demanda, ao invés da constante troca por equipamentos mais potentes (*scale up*). O Monsoon estabelece uma arquitetura de *data center* organizada em formato de malha com o objetivo de comportar 100.000 ou mais servidores. Para criar esta malha são utilizados *switches* programáveis de camada 2 (*comoditizados*) e servidores. Modificações no plano de controle dos equipamentos são necessárias para suportar, por exemplo, roteamento com rota na origem. As modificações no plano de dados, por sua vez, visam suportar o encaminhamento de dados por múltiplos caminhos através do *Valiant Load Balancing* (VLB). Nesta proposta, a função de balanceamento de carga é compartilhada por um grupo de servidores. Consequentemente, os equipamentos responsáveis pelo balanceamento de carga podem ser distribuídos pelos *racks* do *data center*, oferecendo maior agilidade e menor fragmentação na utilização dos recursos disponíveis.

3.3.1.1. Arquitetura

Os aspectos principais do Monsoon podem ser definidos como: (1) engenharia de tráfego utilizando uma malha de camada 2; (2) escalabilidade para criar grandes domínios de camada 2 e (3) espalhamento de carga ubíquo[11].

A Figura 3.16 apresenta uma visão geral da arquitetura do Monsoon. Os dois principais aspectos da arquitetura são: (1) a definição de uma única rede de camada 2 na qual todos os 100.000 servidores são conectados e (2) a flexibilidade pela qual as requisições podem ser distribuídas entre os diversos conjuntos de servidores.

Se observadas a partir de uma perspectiva arquitetural de mais alto nível, as diferenças entre as camadas 2 (Ethernet) e 3 (IP) estão diminuindo, especialmente para redes contidas dentro de um mesmo prédio. Entretanto, existem alguns fatores práticos que levaram o Monsoon a optar pela camada 2 como tecnologia para criar um único domínio no qual todos os servidores estão conectados: (1) cortar custos; (2) eliminar a fragmentação de servidores e (3) diminuir o distúrbio às aplicações. Considerando estes fatores, o Ethernet é claramente a tecnologia eleita, pois está abaixo do IP e já apresenta custo e desempenho otimizados para o encaminhamento baseado em endereços planos. Atualmente, uma porta Ethernet custa entre

[11]Espalhamento de carga ubíquo, neste contexto, significa distribuir todo o tráfego presente no interior do *data center* através de sua estrutura total de *switches* para melhor aproveitamento da infraestrutura.

10% e 50% do valor de uma porta de velocidade equivalente de camada 3 e um *data center* possui centenas de milhares de portas.

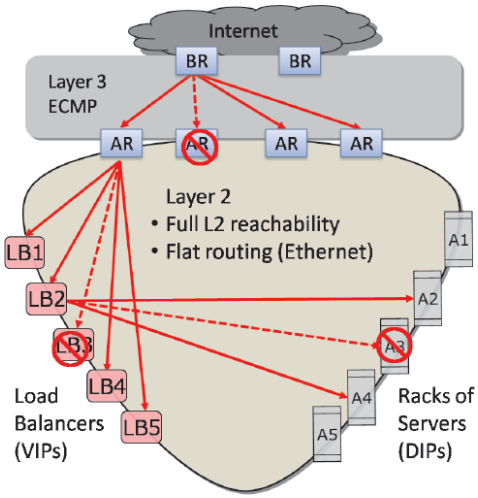


Figura 3.16. Visão geral da arquitetura do Monsoon. Extraída de [Greenberg et al. 2008].

Como observado na Figura 3.16, o domínio de camada 2 provê total conectividade entre os servidores do *data center*. A comunicação entre os servidores utiliza a taxa máxima das interfaces de rede, sem que ocorram enlaces sobrecarregados, ou seja, todos os servidores podem comunicar entre si a 1 Gbps. Ainda na Figura 3.16, a porção da rede que utiliza a camada 3 é necessária para conectar o *data center* à Internet, utilizando roteadores de borda (BRs - *Border Routers*) e o *Equal Cost MultiPath* (ECMP) para espalhar as requisições igualmente entre os roteadores de acesso (ARs - *Access Routers*).

Assim que as requisições adentram o *data center*, os roteadores de acesso utilizam técnicas de *hash* consistente para distribuir as requisições, uniformemente, entre os diversos balanceadores de carga (LBs - *Load Balancers*) que estão associados a um endereço IP virtual (VIP - *Virtual IP*) de uma determinada aplicação visível publicamente. Finalmente, os balanceadores de carga utilizam funções de distribuição específicas de cada aplicação para espalhar as requisições no conjunto de servidores, identificados pelos endereços IP diretos (DIPs - *Direct IPs*), que estão executando a aplicação desejada.

A Figura 3.16 indica a existência de um mecanismo de recuperação de falhas, sejam elas em qualquer dos roteadores de acesso, balanceadores de carga ou servidores. Um serviço de recuperação (*health service*) continuamente monitora o estado de todos os equipamentos que constituem o *data center*. Por exemplo, um servidor que venha a falhar é imediatamente removido do conjunto de servidores associados a uma determinada aplicação, evitando que novas requisições sejam encaminhadas para o servidor em falha.

3.3.1.2. Encaminhamento Servidor-a-Servidor

O Monsoon utiliza tunelamento MAC-in-MAC para encaminhar pacotes entre os servidores que constituem o *data center*. Para tanto, é utilizado um serviço de diretório no qual a lista de endereços MAC dos servidores responsáveis pelas requisições, bem como o endereço MAC dos *switches* nos quais os servidores estão conectados, é mantida.

A Figura 3.17 apresenta a pilha de rede implementada pelos servidores do Monsoon. Nesta pilha, a tradicional função de ARP é desativada e substituída por um processo executado em espaço de usuário (*Monsoon Agent*) e uma nova interface MAC virtual (*Encap-*

sulator). Note que estas mudanças são imperceptíveis para as aplicações. Basicamente, o encapsulador recebe pacotes vindos da camada de rede (IP) e consulta seu cache de resoluções MAC em busca da informação de encaminhamento. Caso não exista, o encapsulador solicita ao agente uma nova resolução através do serviço de diretório.

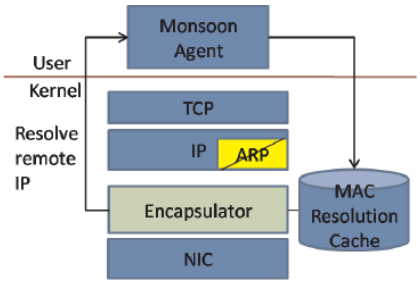


Figura 3.17. Pilha de rede implementada pelos servidores do Monsoon. Extraída de [Greenberg et al. 2008].

O serviço de diretório resolve o endereço IP de destino e retorna uma lista contendo todos os endereços MAC dos servidores associados ao serviço solicitado. Com base nesta lista, o encapsulador escolhe um servidor (seleciona seu MAC), encontra o MAC do *switch* de topo de *rack* (TOR) no qual o servidor está conectado e, finalmente, escolhe um *switch* intermediário no qual os pacotes serão enviados (*bounce off*) por questões de balanceamento de carga do VLB. Este conjunto de informações corresponde a um único fluxo de dados e é mantido em cache para que todos os quadros do fluxo recebam o mesmo tratamento. A Figura 3.18 detalha o encaminhamento dos quadros entre origem e destino.

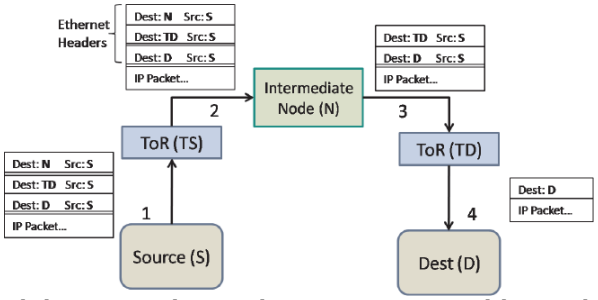


Figura 3.18. Encaminhamento de quadros entre os servidores de origem e destino via encapsulamento MAC-in-MAC. Extraída de [Greenberg et al. 2008].

3.3.1.3. Conectividade externa

A Figura 3.19 apresenta o caminho utilizado pelas conexões que são originadas ou destinadas de/para a Internet. Basicamente, todo o tráfego entra ou sai do *data center* através dos roteadores de borda que, por sua vez, estão conectados a um conjunto de roteadores de acesso utilizando uma rede de camada 3, na qual o ECMP é implementado. Entretanto, os roteadores de acesso não suportam as primitivas de espalhamento e encapsulamento do Monsoon. Sendo assim, cada roteador de acesso possui um servidor de ingresso associado a ele e todo tráfego externo que chega ao roteador de acesso é encaminhado para este servidor de ingresso para receber o tratamento adequado do Monsoon.

Cada servidor de ingresso possui duas interfaces de rede, uma conectada diretamente ao roteador de acesso e a outra conectada à rede do *data center* através de um *switch* TOR. No caso dos pacotes vindos da Internet, o servidor de ingresso utiliza o serviço de diretório para resolver o endereço IP e encaminha o tráfego dentro da rede do *data center* como qualquer outro servidor. Para os pacotes na direção da Internet, o serviço de diretório mapeia o endereço MAC do *gateway default* para o endereço MAC dos servidores de ingresso, possibilitando assim a saída dos pacotes para a Internet.

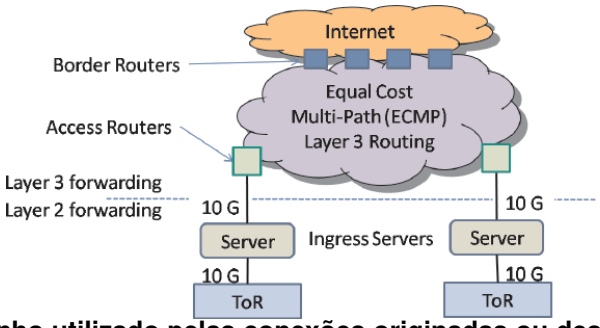


Figura 3.19. Caminho utilizado pelas conexões originadas ou destinadas à Internet. Extraída de [Greenberg et al. 2008].

3.3.1.4. Topologia de *switches*

A Figura 3.20 apresenta um exemplo concreto de uma topologia capaz de conectar 103.680 servidores em um único domínio de camada 2, na qual o VLB seria bem suportado. Neste cenário, cada *switch* TOR possui 2 portas Ethernet de 10Gbps que são conectadas a dois *switches* diferentes de ingresso/egresso localizados na região central (*core*) do *data center*, por questões de tolerância a falhas.

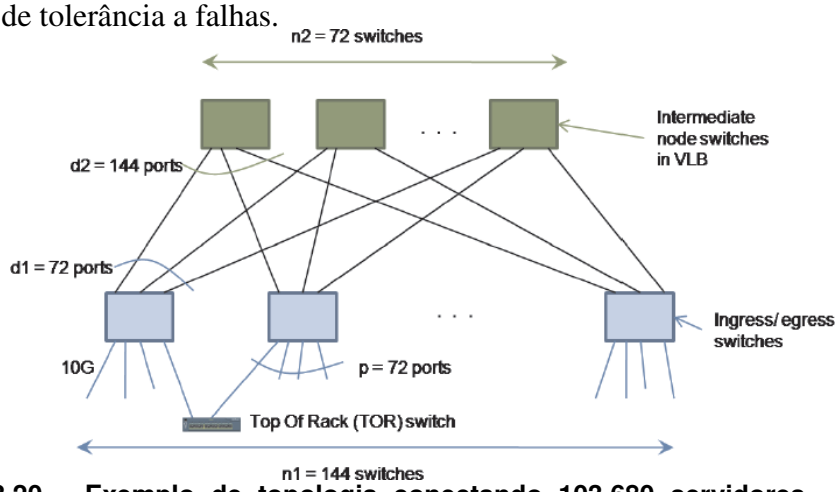


Figura 3.20. Exemplo de topologia conectando 103.680 servidores. Extraída de [Greenberg et al. 2008].

A região central é organizada em dois níveis (*n1* e *n2*) de *switches*. Neste exemplo, o nível *n1* possui 144 *switches*, representados pelos *switches* cinza claro, que não possuem qualquer ligação entre eles, mas cada um deles está conectado a todos os *switches* intermediários (nível *n2*) através de portas Ethernet de 10 Gbps. Sendo assim, existem 72 *switches* no nível *n2* (intermediários), representados na figura em cinza escuro, tornando a topologia interessante para o VLB, pois os fluxos podem escolher os *switches* intermediários nos quais eles irão alcançar a partir deste conjunto com 72 *switches*.

Finalmente, considerando-se o número de portas e *switches* presentes no nível *n1*, esta topologia é capaz de conectar 5184 *racks* com 20 servidores cada, totalizando 103.680 servidores em um único domínio de camada 2, possibilitando que cada servidor transmita à taxa máxima de sua interface (1 Gbps).

3.3.2. VL2

O VL2 é uma proposta de nova arquitetura de *data center* e pertence ao mesmo grupo de pesquisa da Microsoft que originou o Monsoon. Sendo assim, o VL2 pode ser considerado como uma evolução do Monsoon, introduzindo alguns refinamentos.

O VL2 [Greenberg et al. 2009b] trata as limitações citadas na Seção 3.2.6 através da criação de uma camada 2 virtual (VL2 - *Virtual Layer 2*). Esta camada virtual provê aos serviços do *data center* a ilusão de que todos os servidores associados a eles, e apenas os servidores associados a eles, estão conectados através de um único *switch* de camada 2 (livre de interferências) e mantém esta ilusão mesmo que o tamanho de cada serviço varie entre 1 ou 100.000 servidores. Para atingir esta meta, é preciso construir uma rede que honre três objetivos: (1) a comunicação servidor-a-servidor só pode ser limitada pela taxa de transmissão das interfaces de rede de cada servidor (1 Gbps); (2) o tráfego gerado por um serviço deve ser isolado de tal forma a não afetar o tráfego de outros serviços e (3) o *data center* deve ser capaz de alocar qualquer servidor para atender a qualquer serviço (agilidade), possibilitando a atribuição de qualquer endereço IP àquele servidor, de acordo com as exigências do serviço e independente da sua localização no *data center*.

Além de propor uma arquitetura que busca prover agilidade na alocação de servidores, o VL2 investiga que tipo de solução poderia ser construída utilizando os recursos disponíveis atualmente, evitando qualquer tipo de modificação no hardware de *switches* e servidores, oferecendo ainda, um cenário transparente para aplicações legadas. Sendo assim, uma prática comum em *data centers* é a modificação do software (sistemas operacionais) utilizados nos servidores. Neste contexto, o VL2 propõe uma reorganização nos papéis desempenhados tanto pelos servidores quanto pela rede, através da introdução de uma camada de software (*shim*) na pilha de protocolos implementada pelos servidores, de tal forma a contornar as limitações impostas pelos dispositivos legados de rede.

3.3.2.1. Arquitetura

A Figura 3.21 apresenta a topologia utilizada pelo VL2, um *backbone* com elevada conectividade entre os *switches* de agregação e intermediários. Os *switches* ToR são conectados a dois *switches* de agregação e, devido ao elevado número de conexões disponível entre qualquer par de *switches* de agregação, a falha de qualquer um dos n *switches* intermediários reduz em apenas $1/n$ a largura de banda disponível, garantindo uma lenta degradação do serviço oferecido pelo *data center*.

A rede é constituída por duas classes de endereços, os endereços com significado topológico (LAs - *Locator Addresses*) e os endereços planos de aplicação (AAs - *Application Addresses*). Neste cenário, a infraestrutura de rede utiliza endereços LA, que são atribuídos para todos os *switches* e suas interfaces. Além disso, todos os *switches* executam um protocolo de roteamento baseado no estado do enlace para disseminar estes LAs, oferecendo uma visão global da topologia formada pelos *switches* e possibilitando o encaminhamento de pacotes encapsulados em LAs através de caminhos mais curtos. Por outro lado, as aplicações utilizam AAs que permanecem inalterados, independentemente da maneira como os servidores migram no interior do *data center*. Para todo AA é associado o LA atribuído ao *switch* ToR no qual o servidor está conectado. Este mapeamento é mantido por um serviço de diretório do VL2.

A malha de camada 3 formada pelo VL2 cria a ilusão de um único domínio de camada 2 para os servidores no interior do *data center*, uma vez que os servidores imaginam pertencer a uma mesma VLAN. Note que todos os servidores na Figura 3.21 possuem um endereço AA alocado a partir da faixa 20/8. Neste cenário de camada 3, as requisições vindas da Internet podem fluir diretamente até os servidores, sem serem forçadas através

de *gateways* específicos nos quais os pacotes são re-escritos como ocorre, por exemplo, no Monsoon [Greenberg et al. 2008]. Para tanto, endereços LA adicionais são atribuídos aos servidores a partir de uma faixa de IPs válidos (alcançáveis) na Internet.

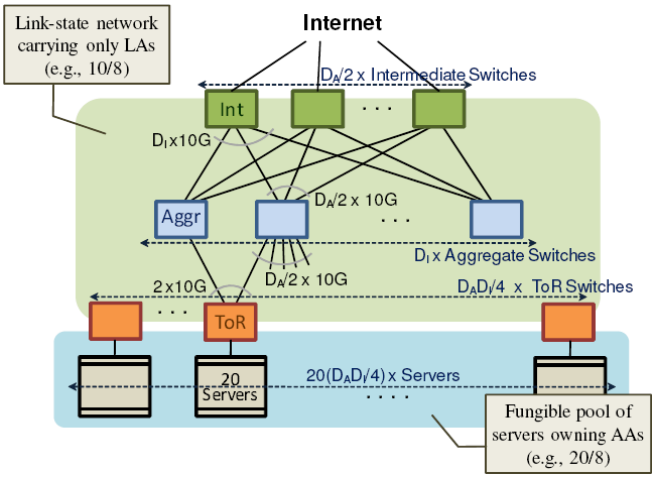


Figura 3.21. Exemplo de *backbone* utilizado pelo VL2. Extraída de [Greenberg et al. 2009b].

3.3.2.2. Endereçamento e roteamento

A Figura 3.22 apresenta um exemplo de encaminhamento de pacotes através da estrutura do VL2. Basicamente, para rotear tráfego entre servidores identificados por endereços AA em uma rede que possui rotas formadas a partir de endereços LA, um agente VL2 executando em cada um dos servidores intercepta os pacotes originados e os encapsula em pacotes endereçados ao LA do *switch* ToR associado ao servidor de destino. Existe ainda um terceiro cabeçalho encapsulando todos os pacotes por razões de espalhamento de tráfego que será detalhado na Seção 3.3.2.3. Basicamente, o sucesso do VL2 está associado ao fato dos servidores acreditarem compartilhar uma única sub-rede IP, devido ao encapsulamento efetuado.

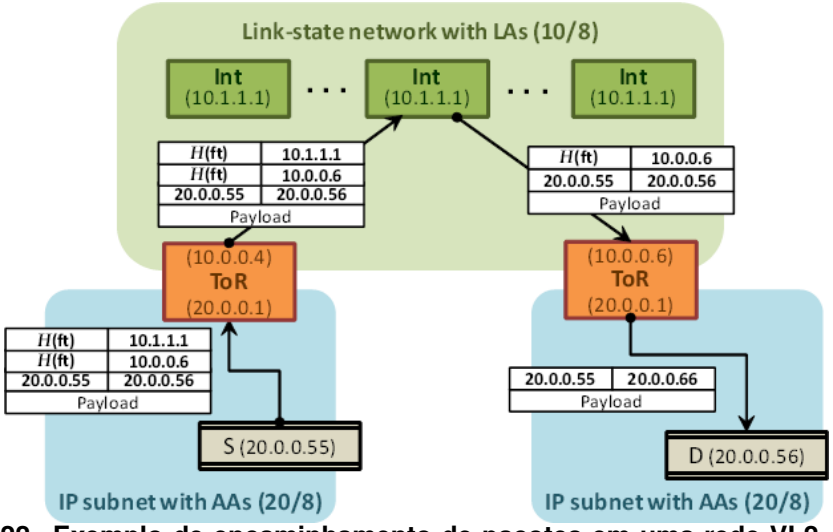


Figura 3.22. Exemplo de encaminhamento de pacotes em uma rede VL2. Extraída de [Greenberg et al. 2009b].

Como forma de contornar o *broadcast* gerado pelo protocolo ARP, na primeira vez em que um servidor envia pacotes para um determinado endereço AA, a pilha de rede original do servidor gera uma requisição ARP e a envia para a rede. Neste instante, o agente VL2

presente no servidor intercepta a requisição ARP e a converte em uma pergunta *unicast* para o serviço de diretório do VL2. O serviço de diretório, por sua vez, responde com o endereço LA do ToR de destino no qual os pacotes devem ser tunelados e o agente VL2 armazena este mapeamento, em um procedimento similar ao cache original do ARP, evitando novas perguntas ao serviço de diretório.

3.3.2.3. Espalhamento de tráfego por múltiplos caminhos

O VL2 combina o VLB e o ECMP como forma de evitar áreas de concentração de tráfego. O VLB é utilizado para distribuir o tráfego entre um conjunto de *switches* intermediários e o ECMP é utilizado para distribuir o tráfego entre caminhos de custo igual. A combinação de ambos os mecanismos cria um cenário de distribuição de tráfego mais eficaz, uma vez que as limitações presentes em um mecanismo são tratadas pelo outro.

A Figura 3.22 ilustra como o agente VL2 utiliza o encapsulamento para implementar o VLB e enviar o tráfego através de um *switch* intermediário escolhido aleatoriamente. Em suma, entre a origem e o destino, o pacote é enviado aos *switches* intermediários, desencapsulado por este *switch*, encaminhado para o ToR de destino, desencapsulado novamente e, finalmente, enviado ao destinatário. Este processo de encapsulamento de pacotes na direção de um *switch* intermediário satisfaz o VLB. Entretanto, eventuais falhas nestes *switches* intermediários poderiam levar a um cenário no qual um elevado número de agentes VL2 teriam de ser atualizados para convergir ao novo estado da rede.

O VL2 contorna esta situação atribuindo o mesmo endereço LA para todos os *switches* intermediários (10.1.1.1 na Figura 3.22). Note que na topologia adotada pelo VL2, todos os *switches* intermediários estão a exatos três saltos de distância dos servidores de origem, criando o cenário adequado para a utilização do ECMP. Sendo assim, o ECMP assume a responsabilidade de entregar os pacotes para um dos *switches* intermediários e, em caso de falhas, o ECMP reage, enviando os pacotes para um *switch* que esteja operacional, eliminado a necessidade de avisar os diversos agentes VL2.

3.3.2.4. Serviço de diretório

O serviço de diretório do VL2 provê três serviços principais: (1) consultas; (2) atualizações de mapeamentos entre AAs e LAs e (3) um mecanismo de atualização de cache reativo para atualizações sensíveis a atrasos (por exemplo, a atualização entre um AA e um LA durante o processo de migração de uma máquina virtual).

Considerando os requisitos de desempenho e os padrões de consultas e atualizações, o VL2 define uma arquitetura de dois níveis conforme ilustra a Figura 3.23. O primeiro nível, considerando-se um *data center* com aproximadamente 100.000 servidores, possui entre 50 e 100 servidores de diretório (DS - *Directory Servers*) otimizados para leitura (consultas), utilizados para replicar os dados do serviço de diretório e responder a consultas dos agentes VL2. No segundo nível, existe um número pequeno, entre 5 e 10, de servidores otimizados para escrita (RSM - *Replicated State Machine*), que oferecem um serviço consistente de armazenamento.

Cada servidor DS possui em cache todos os mapeamentos AA-LA disponíveis nos servidores RSM e responde às consultas feitas pelos agentes VL2 de forma independente. Para estes servidores DS, não há a necessidade do oferecimento de elevados níveis de consistência. Sendo assim, as sincronizações com os servidores RSM ocorrem a cada 30 segun-

dos. Em caso de atualizações na rede, o sistema envia a nova informação para um servidor DS que, por sua vez, encaminha a atualização para um servidor RSM. O servidor RSM replica esta informação entre todos os servidores RSM e, finalmente, envia uma mensagem de confirmação para o servidor de diretório que originou a atualização.

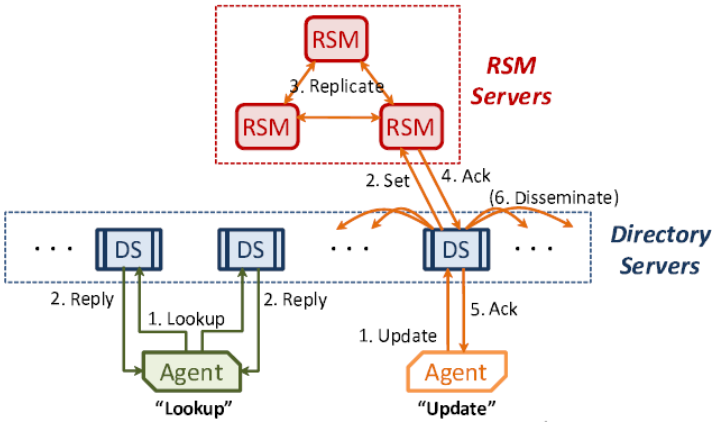


Figura 3.23. Arquitetura do serviço de diretório do VL2. Extraída de [Greenberg et al. 2009b].

3.3.3. Portland

Esta seção introduz o PortLand [Mysore et al. 2009], um conjunto de protocolos compatíveis com Ethernet para efetuar roteamento, encaminhamento e resolução de endereços. O PortLand é desenvolvido considerando-se a estrutura organizacional comumente encontrada em *data centers*, ou seja, uma árvore com múltiplos nós na raiz (denominada *fat tree*). O Portland pode ser considerado como uma versão mais viável e refinada da proposta anterior de um dos autores [Al-Fares et al. 2008], baseada também em uma topologia *fat-tree* e *switches comoditizados*, mas fundamentada em um esquema de roteamento IP customizado.

Com base neste cenário, o PortLand propõe: (1) a utilização de um protocolo para possibilitar que os *switches* descubram sua posição (topológica) na rede; (2) a atribuição de Pseudo endereços MAC (PMAC) para todos os nós finais, de forma a codificar suas posições na topologia; (3) a existência de um serviço centralizado de gerenciamento da infraestrutura de rede (*Fabric Manager*) e (4) a implantação de um serviço de *Proxy* para contornar o *broadcast* inerente ao ARP.

3.3.3.1. Arquitetura

A Figura 3.24 ilustra uma topologia *fat tree* em três níveis utilizada pelo PortLand. Para construir uma topologia em três níveis como esta, é preciso estabelecer o parâmetro k que define o número de portas em cada *switch* (k=4 neste exemplo). Em geral, a topologia em três níveis constituída de *switches* com k portas suporta comunicação não-bloqueante entre $k^3/4$ servidores utilizando $5k^2/4$ *switches* de k portas. A topologia como um todo é organizada em k conjuntos de servidores (chamados de *pods*), nos quais é possível prover comunicação não-bloqueante entre $k^2/4$ servidores através de técnicas de *hash* e distribuição do ECMP.

Do ponto de vista organizacional, a rede *fat tree* é relativamente fixa, possibilitando a construção e manutenção de *data centers* modulares, nos quais a filosofia de expansão é adicionar mais equipamentos (*racks* ou colunas de servidores) sob demanda. Obviamente, toda expansão requer uma fase anterior de planejamento na qual a estrutura de *switches* é definida de forma a suportar tais evoluções.

3.3.3.2. *Fabric Manager*

O PortLand utiliza um gerenciador de infraestrutura de rede centralizado (denominado *Fabric Manager*) para manter estados relacionados à configuração da rede, tal como a sua topologia. O *Fabric Manager* é um processo executado no espaço do usuário em uma máquina dedicada responsável pelo auxílio às requisições do ARP, tolerância a falhas e operações de *multicast*. De acordo com a especificação do PortLand, o *Fabric Manager* pode ser desenvolvido como um servidor conectado de forma redundante à estrutura do *data center* ou, ainda, ser executado em uma rede de controle separada.

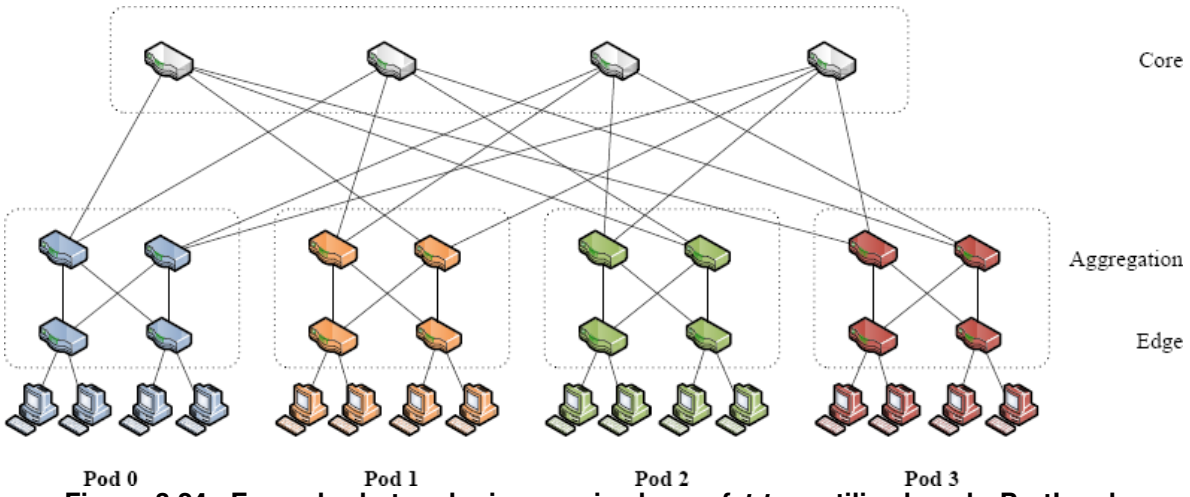


Figura 3.24. Exemplo de topologia organizada em *fat tree* utilizada pelo PortLand. Extraída de [Mysore et al. 2009].

3.3.3.3. Endereços PMAC (Pseudo MAC)

A base para um mecanismo de encaminhamento e roteamento eficiente, bem como suportar a migração de máquinas virtuais no Portland, vem da utilização dos endereços hierárquicos chamados Pseudo MAC (PMAC). O PortLand atribui um único PMAC para cada nó final, representando a localização de cada nó na topologia. Por exemplo, todos os nós finais localizados em um determinado *pod* compartilham um mesmo prefixo em seu PMAC. Entretanto, os nós finais permanecem inalterados, ou seja, eles acreditam ser identificados por seus endereços MAC atuais (AMAC - *Actual MAC*). As requisições de ARP feitas pelos nós finais são respondidas com o PMAC do nó de destino. Sendo assim, todo processo de encaminhamento de pacotes ocorre através da utilização dos PMAC. Os *switches* de egresso são responsáveis pelo mapeamento PMAC para AMAC e re-escrita dos pacotes para manter a ilusão de endereços MAC inalterados no nó de destino.

Os *switches* de borda (*Edge*) aprendem um único número de *pod* e uma única posição dentro deste *pod*. Para todos os nós diretamente conectados, os *switches* de borda atribuem um PMAC de 48 bits sob o formato *pod*.posição.porta.*vmid*, onde *pod* possui 16 bits e refere-se ao número do *pod* onde os nós estão localizados, posição possui 8 bits e indica a posição do *switch* dentro do *pod* e porta possui 8 bits para representar a porta na qual o nó final está ligado ao *switch*. O campo *vmid* possui 16 bits e é utilizado para multiplexar máquinas virtuais em uma mesma máquina física.

No instante em que um *switch* de ingresso observa a existência de um novo endereço MAC, os pacotes com este endereço são desviados para o plano de controle do *switch*, que

cria uma nova entrada na tabela de mapeamento e, na sequência, encaminha este novo mapeamento para o *Fabric Manager* para futuras resoluções, conforme ilustra a Figura 3.25.

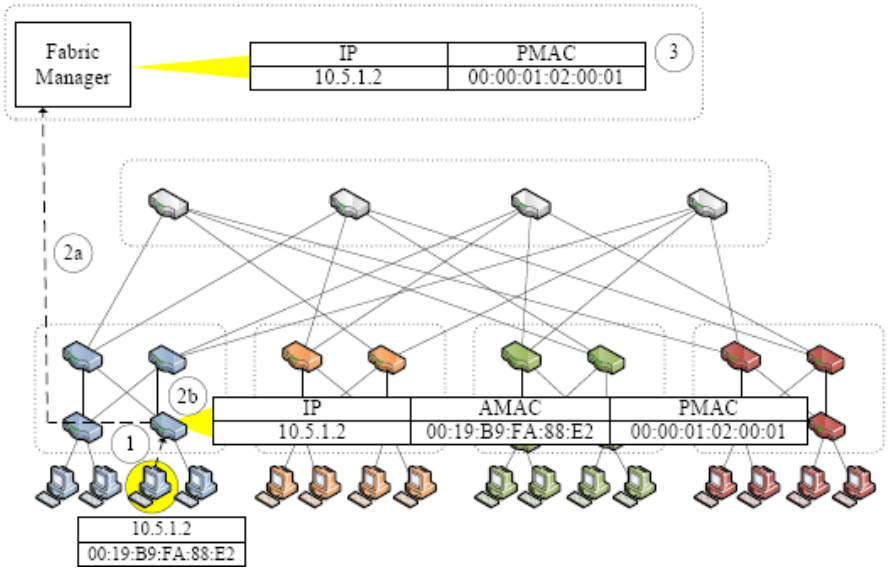


Figura 3.25. Mapeamento entre AMAC e PMAC. Extraída de [Mysore et al. 2009].

Basicamente, o PortLand efetua a separação entre localizador/identificador de forma totalmente transparente aos nós finais e compatível com o hardware dos *switches comoditizados* disponíveis no mercado. Outra característica importante do PortLand refere-se a não utilização de técnicas de tunelamento para encaminhar os pacotes, sendo apenas necessário a re-escrita de endereços PMAC/AMAC nas bordas do *data center*.

3.3.3.4. Mecanismo de *proxy* para requisições ARP

As requisições ARP, originalmente, efetuam *broadcast* e atingem todos os nós localizados em um mesmo domínio de camada 2. O PortLand utiliza o *Fabric Manager* para contornar o *overhead* de sinalização causado pelo ARP conforme ilustra a Figura 3.26. No passo 1, o *switch* de ingresso detecta a chegada de uma mensagem ARP requisitando um mapeamento IP para MAC, intercepta esta mensagem e a encaminha para o *Fabric Manager* no passo 2. O *Fabric Manager* consulta sua tabela de PMACs em busca do mapeamento e retorna o PMAC para o *switch* requisitante no passo 3. O *switch* de borda, por sua vez, cria uma mensagem de resposta do ARP e a retorna para o nó que originou a requisição no passo 4.

Finalmente, existe um detalhe adicional para prover suporte à migração de máquinas virtuais. Assim que a migração é completada, ou seja, a máquina virtual acaba sua transição entre um servidor físico e outro, a máquina virtual envia um ARP gratuito contendo seu novo mapeamento entre endereços IP e MAC. Este ARP gratuito é encaminhado até o *Fabric Manager* pelo *switch* de borda. Infelizmente, os nós que estavam comunicando com esta máquina virtual antes da migração manterão o mapeamento antigo em sua memória cache e terão de esperar até que o mapeamento expire para prosseguir com a comunicação. Entretanto, o *Fabric Manager* pode encaminhar uma mensagem de invalidação de mapeamento ao *switch* no qual a máquina virtual estava associada. Desta maneira, o *switch* seria capaz de replicar o ARP gratuito aos nós que continuam a originar pacotes na direção da máquina virtual que migrou, atualizando seus mapeamentos.

3.3.3.5. Protocolo de descoberta de posição na topologia

Os *switches* utilizam informações relativas às suas posições na topologia global do *data center* para efetuar encaminhamento e roteamento mais eficientes através da comunicação em pares, ou seja, encaminhamento considerando apenas os vizinhos diretamente conectados. Com o objetivo de criar um cenário *plug-and-play*, o PortLand propõe a utilização de um protocolo para descobrir a localização dos *switches* de forma automática.

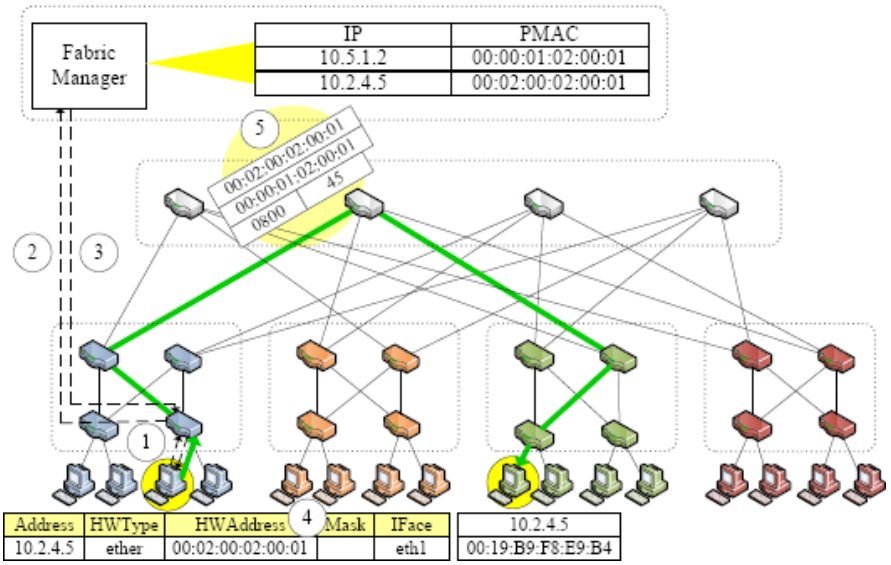


Figura 3.26. Resoluções ARP utilizando mecanismo de *Proxy*. Extraída de [Mysore et al. 2009].

Neste protocolo chamado LDP (*Location Discovery Protocol*), os *switches* enviam, periodicamente, LDMs (*Location Discovery Messages*) em todas as suas portas para: (1) definir suas posições e (2) monitorar o estado de suas conexões físicas. Em resumo, o LDP consegue definir quais são os *switches* de borda (*Edge*), uma vez que eles recebem LDMs em apenas uma fração de suas portas, as conectadas aos *switches* de agregação (*Aggregation*), pois os nós finais não geram LDMs.

A partir do momento que os *switches* de borda descobrem sua localização (nível) na topologia, as LDMs disseminadas subsequentemente passam a conter informação referente ao seu nível. Desta forma, o restante dos *switches* são capazes de aferir suas respectivas posições. Os *switches* de agregação aferem sua posição, uma vez que eles recebem LDMs em todas as suas portas, sendo algumas originadas pelos *switches* de borda (contendo informação de nível) e o restante originadas pelos *switches* de núcleo (sem informação de nível). Finalmente, os *switches* de núcleo (*core*) aferem sua posição, uma vez que todas as suas portas passarão a receber LDMs originadas por *switches* de agregação (contendo informação de nível). Uma vez definido o nível de todos os *switches*, o *Fabric Manager* é utilizado para atribuir o mesmo número de *pod* para os *switches* de borda pertencentes ao mesmo grupo.

3.3.4. BCube e MDCube

Uma nova maneira de construir e implantar *data centers* é através da sua modularização em contêineres, denominados *data centers* modulares (MDCs - *Modular data centers*). Em um MDC, alguns milhares de servidores são conectados utilizando pequenos *switches* para formar a infraestrutura do *data center* e, então, empacotados em contêineres padronizados de

20 ou 40 pés comumente utilizados para transportes mercantes. Desta forma, os *data centers* não são mais restritos a uma localização fixa, possibilitando que as organizações coloquem seus *data centers* em qualquer lugar e, também, possam realocá-los na medida em que seus requisitos mudam. Além da mobilidade, um MDC possui menor tempo de implantação, oferece um sistema com maior densidade de equipamentos e menores custos de resfriamento e produção. Entretanto, é difícil, ou mesmo impossível, efetuar manutenção no interior do contêiner uma vez que ele está implantado.

Esta seção apresenta o BCube [Guo et al. 2009], uma arquitetura robusta e de alto desempenho para a construção de MDCs e, também, apresenta o MDCube [Wu et al. 2009], uma estrutura para a construção de mega *data centers*. Neste cenário, cada contêiner desenvolvido de acordo com a arquitetura do BCube é uma peça dentro de um mega *data center* organizado segundo a estrutura do MDCube.

A arquitetura do BCube adota o modelo *server-centric*, ou seja, toda a inteligência necessária para o funcionamento de um MDC é inserida nos servidores, possibilitando a utilização de *switches comoditizados*. Cada servidor é equipado com um pequeno número de interfaces de rede (tipicamente não mais do que quatro) e múltiplas camadas de *switches* pequenos (poucas portas) são utilizadas para conectar todos os servidores.

A proposta do MDCube é considerar cada contêiner como um nó virtual dentro de um mega *data center*. Cada contêiner possui um conjunto de portas de alta velocidade que são utilizadas para conectá-lo a outros contêineres, criando uma malha de interconexão constituída por enlaces de fibra ótica de baixo custo. O MDCube ainda encontra-se em fase de desenvolvimento e, basicamente, estende a metodologia desenvolvida no BCube para um cenário de mega *data centers*. Os objetivos principais são: (1) prover elevada capacidade de transmissão entre contêineres; (2) diminuir os custos de infraestrutura e (3) simplificar a estrutura de cabeamento necessária.

3.3.4.1. Arquitetura BCube

Existem dois tipos de dispositivos no BCube: servidores com múltiplas interfaces de rede e *switches* que se conectam a um número (pequeno) constante de servidores. A estrutura do BCube é definida através de recursividade, um BCube_0 nada mais é que um conjunto de n servidores conectados a um *switch* de n portas. Um BCube_1 é constituído por n BCubes_0 e n *switches* de n portas. De forma genérica, um BCube_k ($k \geq 1$) é constituído de n BCubes_{k-1} e n^k *switches* de n portas. Cada servidor em um BCube_k possui $k + 1$ portas, as quais são numeradas a partir do nível 0 até o nível k. Com base nestas definições, é trivial perceber que um BCube_k possui $N = n^{k+1}$ servidores e $k + 1$ níveis de *switch*, onde cada nível possui n^k *switches* de n portas.

A Figura 3.27 apresenta um BCube1 com $n = 4$, constituído por quatro BCubes_0 e quatro *switches* de 4 portas. Todos os enlaces do BCube são bidirecionais e a construção garante que os *switches* apenas se conectam a servidores, ou seja, *switches* nunca se conectam a outros *switches* diretamente. A título de exemplificação, utilizando *switches comoditizados* de 8 portas, é possível construir um BCube_3 com 4096 servidores.

3.3.4.2. Mecanismo de Rota na Origem do BCube

O BCube requer um protocolo de roteamento capaz de utilizar de forma eficiente a diversidade de caminhos disponíveis na sua topologia, efetuando uma distribuição de carga ade-

quada. Sendo assim, é proposto o *BCube Source Routing* (BSR). Basicamente, no BSR o servidor de origem decide qual o caminho que o fluxo de pacotes deve atravessar através de mensagens de teste (*probing*) transmitidas na rede e, consequentemente, a rota na origem é inserida no cabeçalho dos pacotes. Duas razões foram consideradas para efetuar a opção pelo mecanismo de rota na origem: (1) o servidor de origem escolhe toda a rota para os pacotes sem a intervenção dos servidores intermediários e (2) os servidores intermediários apenas efetuam o encaminhamento dos pacotes com base na rota disponível nos cabeçalhos. Finalmente, a utilização de mensagens de teste (*probing*) cria um cenário no qual a topologia da rede é descoberta reativamente, evitando a utilização de protocolos baseados no estado do enlace, que sofrem de problemas de escalabilidade comumente associados ao *broadcast*.

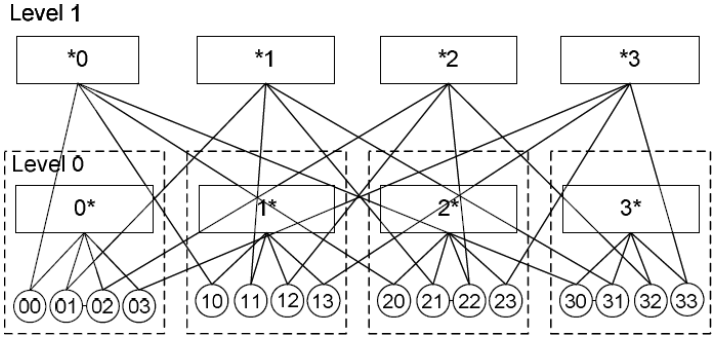


Figura 3.27. Exemplo de um BCube1 com *n* = 4. Extraída de [Guo et al. 2009].

Um fluxo é uma cadeia de pacotes identificados por cinco informações (origem, porta de origem, destino, porta de destino, protocolo). Quando um novo fluxo precisa ser transmitido, a origem envia mensagens de teste através de múltiplos caminhos e os servidores no caminho processam estas mensagens para verificar as necessidades do fluxo. Por exemplo, verificar se atendem as exigências referentes à largura de banda. Assim que as mensagens de teste atingem o destino, ele envia a mensagem de volta ao servidor de origem que, por sua vez, decide que caminho utilizar, baseando-se, por exemplo, na largura de banda disponível.

3.3.4.3. Roteando para redes externas no BCube

O BCube suporta comunicação externa com a Internet e, também, com outros BCubes, propondo a utilização de agregadores e *gateways*. Um agregador é um *switch comoditizado* de camada 2 com interfaces de 10 Gbps e os *gateways* são os servidores conectados nestes *switches* agregadores. Quando um servidor interno envia pacotes para um endereço IP externo, ele escolhe um dos *gateways* disponíveis. O pacote então é roteado para este *gateway* utilizando o BSR e, assim que o *gateway* recebe o pacote, ele remove o cabeçalho com a rota do BSR e encaminha o pacote para a rede externa através das interfaces de 10Gbps dos *switches* agregadores. A comunicação a partir das redes externas atingem o BCube através da utilização de endereços IPs dos servidores, externamente visíveis. Assim que os pacotes adentram o *data center*, os *gateways* são responsáveis pela definição da rota pela qual os pacotes atingirão os servidores.

3.3.4.4. BCube parcial

Em alguns casos, pode ser difícil ou desnecessário a construção de um BCube completo devido a restrições de espaço ou mesmo financeiras. Por exemplo, quando temos n=8 e k=3, é possível inserir 4096 servidores em um BCube_3. Para efetuar a construção de um BCube parcial, todos os BCubes_{k-1} são criados e, então, estes BCubes_{k-1} são conectados utilizando

uma camada k completa de *switches*. Sendo assim, o BSR é capaz de funcionar como se o BCube estivesse completo. A desvantagem desta abordagem é que os *switches* presentes na camada k (completa) não serão totalmente utilizados.

3.3.4.5. Arquitetura do MDCube

O MBCube é desenvolvido para conectar diversos BCubes através das interfaces de alta velocidade disponíveis nos *switches* de agregação destes elementos. Para suportar centenas de contêineres em um mega *data center* e prover uma vazão eficiente de dados, o MDCube utiliza fibra ótica na construção destes enlaces entre BCubes. Neste cenário, as interfaces de alta velocidade disponíveis nos diversos BCubes são vistas como uma única interface virtual. Por exemplo, um *switch comoditizado* que possua quatro interfaces de 10 Gbps é visto como sendo uma interface virtual com a capacidade de 40 Gbps. De acordo com a especificação do MDCube, um BCube é visto como um nó dentro da estrutura de um mega *data center* e o número de interfaces virtuais contidas neste nó é definido pelo número de *switches* de agregação contidos no BCube.

Para a construção de um MDCube $(D + 1)$ dimensional, temos $M = \Pi_{d=0}^{D} m_d$, onde m_d é o número de contêineres em uma dimensão d. Cada contêiner é identificado por um $cid = c_D c_{D-1} ... c_0 (c_d \in [0, m_d - 1], d \in [0,D])$. Cada contêiner armazena $\Sigma_{d=0}^{D}(m_d - 1)$ *switches*, sendo $(m_d - 1)$ o número de *switches* em uma dimensão d. Dentro de um MDCube, cada *switch* é identificado através do seu contêiner ID e seu *switch* ID dentro do BCube: cid, $bwid$, $cid \in [0, M - 1]$, $bwid \in [0, \Sigma_{d=0}^{D}(m_d - 1) - 1]$. Existe um enlace para cada par de contêineres em uma dimensão d. A construção de um MDCube de 2 dimensões é exemplificada na Figura 3.28 para um cenário formado por nove BCubes.

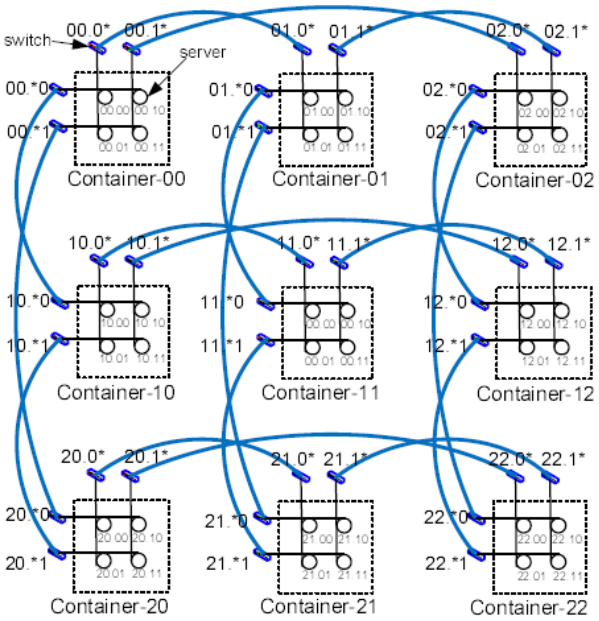


Figura 3.28. Exemplo de um MDCube de 2 dimensões formado a partir de nove (3x3) BCubes. Extraída de [Wu et al. 2009].

O modelo de comunicação do MDCube é *server-centric* como no BCube e a estrutura de roteamento é estendida para contemplar o mega *data center*. Sendo assim, cabe ao servidor de origem enviar as mensagens de teste através da estrutura do mega *data center* de tal forma a descobrir as rotas até o servidor de destino. O roteamento entre os diversos contêineres é baseado na correção das tuplas que identificam os contêineres em cada nó

atravessado pelos pacotes. Esta correção é controlada por uma permutação Π_D e uma nova função introduzida nos servidores possibilita o descobrimento dos enlaces existentes entre os diversos contêineres. A mesma abordagem do BCube para a comunicação externa é utilizada no MDCube. Entretanto, no MDCube as interfaces de alta velocidade dos contêineres são conectadas a roteadores responsáveis pela agregação de tráfego na entrada ou na saída do mega *data center*.

3.3.5. Resumo comparativo das abordagens

A Tabela 3.1 apresenta, de forma compilada, uma análise comparativa das propostas descritas nesta seção, considerando alguns dos requisitos levantados na Seção 3.2.

Tabela 3.1. Análise comparativa das novas arquiteturas de *data center*.

	Monsoon	VL2	Portland	BCube e MDCube
Realocação dinâmica de servidores (agilidade)	sim	sim	sim	sim
Transmissão à taxa máxima das interfaces/ *oversubscription*	sim (1Gbps) / 1:1	sim (1Gbps) / 1:1	sim (1Gbps) / 1:1	sim (1Gbps) / 1:1
Topologia	*fat tree*	*fat tree*	*fat tree*	hipercubo
Mecanismo de Roteamento/ Encaminhamento	tunelamento MAC-in-MAC	tunelamento IP-in-IP	baseado na posição hierárquica dos nós (PMAC)	rota na origem gerada por mensagens de *probing*
Balanceamento de Carga	VLB + ECMP	VLB + ECMP	Não especificado	trocas periódicas de *probing* modificam a rota
Modificação nos Nós Finais	sim	sim	não	sim
Modificação nos *Switches*	sim	não	sim	não
Serviço de diretório	sim	sim	sim	não

3.4. Tendências e conclusões

3.4.1. Tendências

Nesta seção destacamos de forma resumida uma série de tendências tecnológicas e áreas de pesquisa relevantes para o desenvolvimento de *data centers* e a evolução do modelo de computação em nuvem, não necessariamente restritas aos aspectos de arquitetura de rede.

- ***Software-Defined Networking* com OpenFlow:** o paradigma OpenFlow [McKeown et al. 2008] propõe uma generalização do plano de dados que

habilita a programabilidade e a virtualização dos recursos de rede, permitindo a separação entre o plano de dados e plano de controle. O OpenFlow permite que as decisões de manipulação de pacotes em alto-nível sejam realizadas por um controlador separado e centralizado fazendo com que o plano de dados possa ser implementado em *switches comoditizados*, sem possuir uma pilha de protocolos complexa.

Esta proposta tecnológica é especialmente interessante para as redes de *data centers*, de forma que os próprios provedores possam definir a funcionalidade da rede, com um baixo custo do equipamento e, especialmente customizado para as necessidades de escala e controle das características de *data centers* para computação em nuvem. Portland é um dos exemplos apresentados neste minicurso que usa a tecnologia OpenFlow. Espera-se que novas propostas (incluindo também os grandes provedores como Microsoft e Google) adotem a tecnologia OpenFlow (já) para as redes de *clusters* no *data center* [Tavakoli et al. 2009].

Basicamente, um *switch* OpenFlow separa o encaminhamento rápido de pacotes (plano de dados) do nível de decisões de encaminhamento (plano de controle) em um roteador ou *switch*. Embora parte do plano de dados ainda resida no *switch* e execute sobre o mesmo hardware (portas lógicas, memória), as decisões de manipulação de pacotes em alto-nível são movidas para um controlador separado (por exemplo, NOX [Gude et al. 2008]). A Figura 3.29 mostra uma abstração de uma rede com OpenFlow e um controlador NOX.

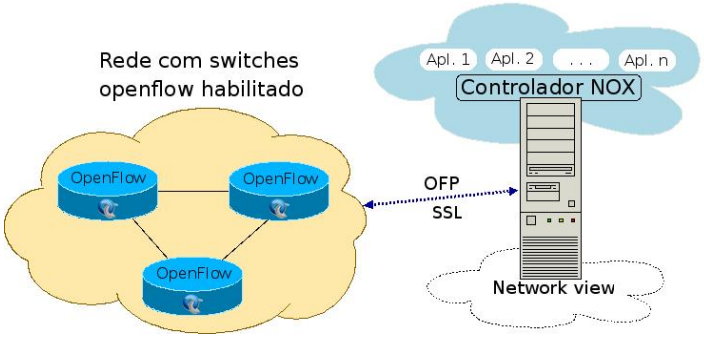


Figura 3.29. Uma rede com OpenFlow e controlador NOX.

- **Arquiteturas baseadas em memória:** com o objetivo de superar o tempo de acesso lento de discos e aumentar o desempenho das aplicações em nuvem, muitos esforços adicionam uma camada de cache e tentam manter o banco de dados e todas as aplicações em sintonia com o cache. A pergunta natural neste sentido é se realmente essa nova camada extra de cache é necessária quando os dados poderiam muito bem ser mantidos na memória desde o início. Com a proliferação de aplicações com requisitos de "tempo real" e sistemas que requerem escalabilidade massiva, os avanços em hardware e software sugerem que a memória volátil pode se tornar o novo disco rígido e os discos rígidos serem relegados às tarefas de *back-up* (tipo fitas).

  Este modelo já foi discutido na comunidade das grades computacionais, onde o avanço em hardware e redes em torno do acesso à memória rápida têm permitido a criação de *clusters* com melhor desempenho do que *clusters* baseados em discos. A memória volátil (por exemplo, RAM) é várias ordens de grandeza mais rápida do que um acesso aleatório em disco.

A fim de preencher a enorme lacuna entre a latência de memória RAM e os discos, novos sistemas de armazenamento (*flash*) SSDs (*Solid-State Drive*), estão sendo introduzidos pelos vendedores. Com a tecnologia *flash* continuando a tendência de redução de preço, esta pode até mesmo substituir as unidades tradicionais para algumas aplicações. Vários projetos já estão trabalhando nessa linha (com ênfase em sistemas de baixo consumo), tais como FAWN [Andersen et al. 2009], Gordon [Caulfield et al. 2009], ou o projeto RAMCloud [Ousterhout et al. 2009] da Universidade de Stanford.

- **Bases de dados NoSQL:** ao longo dos últimos anos, temos assistido o surgimento de sistemas de armazenamento de dados que diferem de forma bastante significativa em relação ao modelo de RDBMS. Estes sistemas são também conhecidos como NoSQL [Leavitt 2010] e entre os exemplos mais notáveis se encontram BigTable da Google, o Dynamo da Amazon, Cassendra, CouchDB, MongoDB, etc.

  Estas soluções têm um número de características em comum, tais como armazenamento de pares chave-valor, a execução em um grande número de máquinas *comoditizadas*, a divisão e replicação dos dados entre as máquinas e o relaxamento do requisito de consistência dos dados. O modelo de dados subjacente pode ser considerado como uma grande tabela *Hash* onde a forma básica de acesso à API é tipicamente da forma *Get (key), Put (key, value), Delete (key)*. Embora os sistemas NoSQL não substituirão os sistemas RDBMS tradicionais, não há dúvida que eles terão um papel importante em muitas das aplicações em nuvem. Nesse sentido, ainda são muitos os desafios que devem ser resolvidos como, por exemplo, o aperfeiçoamento de mecanismos para garantir a confiabilidade, a consistência dos dados e a resiliência a falhas. Vale a pena notar que há interessantes oportunidades de pesquisa na combinação de sistemas NoSQL e as arquiteturas baseadas em memória, pois ambas formam uma parceria natural para os requisitos das aplicações de grande escala nos *data centers* da nuvem.

- **Padrões para computação em nuvem:** a computação em nuvem se insere na Internet como uma nova camada de sistemas distribuídos com APIs fornecidas por múltiplos provedores. Como toda nova camada, oferece vantagens pela abstração dos recursos e oportunidades de inovação. Porém, também requer trabalhos em conjunto para garantir a interoperabilidade das soluções e permitir uma evolução saudável do sistema global como um todo. Vint Cerf, um dos pioneiros da Internet original, tem advertido [Cerf 2010] sobre essa necessidade e aponta os passos que a indústria deve dar para atingir a meta de uma interoperabilidade global na nuvem. Na mesma direção, o *Open Cloud* Manifesto [Open Cloud Manifesto 2010] é uma declaração de princípios para a manutenção da computação em nuvem como um sistema aberto, atualmente apoiada por mais de 250 organizações. Há ainda a iniciativa de padrões para computação em nuvem [Cloud Standards 2010] que atua em conjunto com o *Open Cloud* Manifesto na busca pela definição de padrões para este novo modelo de interação.

- **Interação entre nuvens (*Inter-Cloud*):** até o momento, temos focado nossa discussão nos *data centers* operados de forma isolada, sem entrar em detalhes de cenários (1) onde múltiplos *data centers* operam de forma unificada (por exemplo nos modelos de micro- e nano-*data centers*); (2) onde múltiplos provedores de serviços em nuvem estabelecem acordos para se federar ou (3) onde o cliente (corporativo) dos serviços em

nuvem tem relacionamento com múltiplos provedores (conhecido como nuvem *multi-homing*). Os trabalhos de padronização discutidos anteriormente são um pré-requisito para os cenários 2 e 3. Para tornar estes cenários realidade, ainda há muitos outros desafios e oportunidades para serem abordados, quando analisamos o que pode se chamar de *Inter-Cloud*.

Para começar, podemos imaginar que na comunicação entre nuvens deverá haver um suporte natural do movimento massivo de cargas de trabalho na forma de máquinas virtuais migrando junto a um grande volume de dados. Esta migração pode ocorrer, por exemplo, quando houver a necessidade de aumento de escalabilidade de serviços populares sob-demanda ou a oferta de um provedor oferecendo descontos de recursos computacionais, fazendo com que esta migração entre nuvens seja economicamente rentável. Nestes casos de uso, a comunicação entre nuvens deve garantir segurança dos dados nas múltiplas dimensões conhecidas (integridade, confiabilidade, etc.), transparência nos sistemas de gerência e flexibilidade nos sistemas de endereçamento e roteamento IP/Ethernet de forma que a migração aconteça de modo natural, tanto nas comunicações com as redes virtuais privadas estabelecidas quanto nas comunicações com usuários da Internet pública. A criação de nuvens privadas virtuais [Wood et al. 2009] (fatias de redes virtuais) pode ser realizada como uma rede *overlay* sobre IP ou como uma rede *underlay* (por exemplo mediante o estabelecimento de circuitos óticos multi-domínios) garantindo não só transparência nos protocolos de endereçamento mas também uma grande capacidade de transferência de dados[12].

Finalmente, podemos especular que o desenvolvimento da *Inter-Cloud* pode trazer ainda mais mudanças na infraestrutura central (*core*) da Internet, com novos acordos de *peering* entre os provedores de serviços em nuvem e o surgimento da denominada "*dark Internet*". Também se espera o surgimento de um mercado de recursos na nuvem operado por agregadores e *brokers* (negociadores) para serviços em nuvem (por exemplo, Transit Portal [Valancius et al. 2009]). Serviços de conectividade flexível (multi-homing, balanceamento de carga) e avançada (segurança, QoS) com os provedores de serviços em nuvem podem trazer novos incentivos para a adoção de protocolos, tais como as versões seguras de BGP, DNS, IP *multicast*, MPLS, IPv6 e propostas baseadas na separação identificador/localizador como o LISP.

Em uma evolução extrema da *Inter-Cloud*, pode emergir o conceito de *Ambient Cloud* [Hoff 2009a], motivado pelos custos das mega-infraestruturas e as demandas de aplicações, onde os recursos de qualquer dispositivo (*set-top-box*, PDAs, celulares, PCs) podem ser utilizados em uma escala global.

3.4.2. Conclusões

O modelo convencional de rede IP/Ethernet não atende os requisitos de custo, escala e controle dos provedores de computação em nuvem. É por este motivo e pelas características especiais das redes dos *data centers* que novos projetos e propostas têm emergido para atender os objetivos específicos dos *cloud data centers*, que são criticamente diferentes dos *data centers* tradicionais e das redes locais e metropolitanas dos provedores de serviços.

[12]Estima-se que o volume de tráfego resultante desta migração de máquinas virtuais pode alcançar ordens de magnitude comparáveis à quantidade de vídeo sobre Internet atualmente.

Tratando os *data centers* como um sistema e fazendo uma customização e otimização completa, as novas propostas de arquiteturas de rede prometem atingir uma redução dos custos operacionais e de capital, uma maior confiabilidade, um modelo de escala sob-demanda sustentável e uma maior capacidade de inovação.

Não se pode negar que o modelo de computação em nuvem é evolucionário pois surge de uma construção histórica, baseada na forma como a própria Internet surgiu e cresceu. As demandas por novos serviços e o barateamento de recursos computacionais fizeram com que grandes empresas como Microsoft e Google, que já possuíam um grande patrimônio computacional instalado para atender suas próprias necessidades, percebessem que vender ou alugar tais recursos computacionais poderia ser um negócio rentável. Sendo assim, estas empresas são as que lideram este modelo e continuam investindo cada vez mais para aumentar a disponibilidade de serviços em nuvem. Entretanto, pequenas empresas podem se inserir no mercado como provedores de serviços utilizando uma infraestrutura (IaaS) alugada através do modelo *pay-as-you-go*.

O modelo de computação em nuvem tem atraído vários setores incluindo empresas provedoras de serviços Web, provedores de conectividade, indústria, setor bancário e mais recentemente governos federais de alguns países. Um exemplo disso é o Governo Federal Americano que tem investido no desenvolvimento de soluções em nuvem, utilizando plataformas de código livre (*open source*) para atender às suas próprias necessidades [NASA 2010]. Espera-se que a comunidade brasileira apoiada por órgãos de pesquisa e instituições federais e privadas comece a avaliar as soluções baseadas em serviços em nuvem como uma forma de reduzir custos de TI, aumentar a oferta de serviços e se inserir em um modelo global, inovador e emergente.

Referências

[Al-Fares et al. 2008] Al-Fares, M., Loukissas, A., and Vahdat, A. (2008). A Scalable Commodity Data Center Network Architecture. *SIGCOMM Comput. Commun. Rev.*, 38(4):63–74.

[Amazon 2010a] Amazon (2010a). Amazon Elastic Compute Cloud. Disponível online em http://aws.amazon.com/ec2/.

[Amazon 2010b] Amazon (2010b). Amazon Web Services. Disponível online em http://aws.amazon.com/.

[Andersen et al. 2009] Andersen, D. G., Franklin, J., Kaminsky, M., Phanishayee, A., Tan, L., and Vasudevan, V. (2009). FAWN: A Fast Array of Wimpy Nodes. In *SOSP '09: Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles*, pages 1–14, New York, NY, USA. ACM.

[Armbrust et al. 2009] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R. H., Konwinski, A., Lee, G., Patterson, D. A., Rabkin, A., Stoica, I., and Zaharia, M. (2009). Above the Clouds: A Berkeley View of Cloud Computing. Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley.

[Barroso and Hölzle 2007] Barroso, L. A. and Hölzle, U. (2007). The Case for Energy-Proportional Computing. *Computer*, 40(12):33–37.

[Benson et al. 2009a] Benson, T., Anand, A., Akella, A., and Zhang, M. (2009a). Understanding Data Center Traffic Characteristics. In *WREN '09: Proceedings of the 1st ACM workshop on Research on enterprise networking*, pages 65–72, New York, NY, USA. ACM.

[Benson et al. 2009b] Benson, T. A., Akella, A., and Zhang, M. (2009b). The Case for Fine-Grained Traffic Engineering in Data Centers. Technical Report 1666, University of WisconsinMadison. Disponível online em http://www.cs.wisc.edu/techreports/2009/TR1666.pdf.

[Brandon 2009] Brandon, H. e. a. (2009). ElasticTree: Saving Energy in Data Center Networks. Technical report. Disponível online em http://www.openflowswitch.org/wk/images/2/2c/ElasticTree-TechReport2009.pdf.

[Caulfield et al. 2009] Caulfield, A. M., Grupp, L. M., and Swanson, S. (2009). Gordon: Using Flash Memory to Build Fast, Power-efficient Clusters for Data-intensive Applications. In *ASPLOS '09: Proceeding of the 14th international conference on Architectural support for programming languages and operating systems*, pages 217–228, New York, NY, USA. ACM.

[Cerf 2010] Cerf, V. (2010). Vint Cerf e computação em nuvem. Disponível online em http://news.techworld.com/virtualisation/3209948/vint-cerf-calls-for-cloud-computing-standards/.

[Christian Belady (ed) 2007] Christian Belady (ed) (2007). The Green Grid Data Center Power Efficiency Metrics: PUE and DCiE. Technical report.

[Cisco 2007] Cisco (2007). Cisco Data Center Infrastructure 2.5 Design Guide. Disponível online em http://www.cisco.com/en/US/docs/solutions/Enterprise/Data_Center/DC_Infra2_5/DCI_SRND_2_5_book.html.

[Cloud Standards 2010] Cloud Standards (2010). Padrões para Computação em Nuvem. Disponível online em http://cloud-standards.org.

[Enterprise Control Systems 2010] Enterprise Control Systems (2010). Vendor Neutral Modular and Container Based Data Centers. Disponível online em http://www.datacenterexperts.com/containerbaseddatacenters.html.

[Forrester 2010] Forrester (2010). Should Your Email Live In The Cloud? A Comparative Cost Analysis. Disponível online em http://www.google.com/a/help/intl/en/admins/pdf/forrester_cloud_email_cost_analysis.pdf.

[Foster 2010] Foster, I. (2010). Blog do Ian Foster. Disponível online em http://ianfoster.typepad.com/blog/.

[Google 2010a] Google (2010a). BigTable. Disponível online em http://labs.google.com/papers/bigtable.html.

[Google 2010b] Google (2010b). Google Apps. Disponível online em http://www.google.com/apps/.

[Google 2010c] Google (2010c). Google Web Toolkit. Disponível online em http://code.google.com/intl/pt-BR/webtoolkit/.

[Google 2010d] Google (2010d). Medidas de PUE da Google. Disponível online em http://www.google.com/corporate/green/datacenters/measuring.html.

[Greenberg 2009] Greenberg, A. (2009). Networking The Cloud. ICDCS 2009 keynote. Disponível online em http://www.cse.ohio-state.edu/icdcs2009/Keynote_files/greenbergkeynote.pdf.

[Greenberg et al. 2009a] Greenberg, A., Hamilton, J., Maltz, D. A., and Patel, P. (2009a). The Cost of a Cloud: Research Problems in Data Center Networks. *SIGCOMM Comput. Commun. Rev.*, 39(1):68–73.

[Greenberg et al. 2009b] Greenberg, A., Hamilton, J. R., Jain, N., Kandula, S., Kim, C., Lahiri, P., Maltz, D. A., Patel, P., and Sengupta, S. (2009b). VL2: A Scalable and Flexible Data Center Network. *In Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication, Barcelona, Spain*.

[Greenberg et al. 2008] Greenberg, A., Lahiri, P., Maltz, D. A., Patel, P., and Sengupta, S. (2008). Towards a Next Generation Data Center Architecture: Scalability and Commoditization. *In Proceedings of the ACM Workshop on Programmable Routers For Extensible Services of Tomorrow, Seattle, WA, USA*.

[Greenberg et al. 2006] Greenberg, S., Mills, E., Tschudi, B., Rumsey, P., and Myatt (2006). Best Practices for Data Centers: Results from Benchmarking 22 Data Centers. In *Proceedings of the 2006 ACEEE Summer Study on Energy Efficiency in Buildings.*

[Gude et al. 2008] Gude, N., Koponen, T., Pettit, J., Pfaff, B., Casado, M., McKeown, N., and Shenker, S. (2008). NOX: Towards an Operating System for Networks. *SIGCOMM Comput. Commun. Rev.*, 38(3):105–110.

[Guo et al. 2009] Guo, C., Lu, G., Li, D., Wu, H., Zhang, X., Shi, Y., Tian, C., Zhang, Y., and Lu, S. (2009). BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers. *In Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication, Barcelona, Spain*.

[Guo et al. 2008] Guo, C., Wu, H., Tan, K., Shi, L., Zhang, Y., and Lu, S. (2008). Dcell: A Scalable and Fault-tolerant Network Structure for Data Centers. In *SIGCOMM '08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, pages 75–86, New York, NY, USA. ACM.

[Hamilton 2008] Hamilton, J. (2008). Diseconomies of Scale. Blog post, Disponível online em http://perspectives.mvdirona.com/2008/11/28/CostOfPowerInLargeScaleDataCenters.aspx.

[Hamilton 2009a] Hamilton, J. (2009a). Data Center Networks Are in My Way. *Stanford Clean Slate CTO Summit*.

[Hamilton 2009b] Hamilton, J. (2009b). PUE and Total Power Usage Efficiency (tPUE). Blog post, Disponível online em http://perspectives.mvdirona.com/2009/06/15/PUEAndTotalPowerUsageEfficiencyTPUE.aspx.

[Hamilton 2010a] Hamilton, J. (2010a). Perspectives. Disponível online em http://perspectives.mvdirona.com.

[Hamilton 2010b] Hamilton, J. (2010b). Scaling at MySpace. Blog post, Disponível online em http://perspectives.mvdirona.com/2010/02/15/ScalingAtMySpace.aspx.

[Hoelzle and Barroso 2009] Hoelzle, U. and Barroso, L. A. (2009). *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines*. Morgan and Claypool Publishers.

[Hoff 2009a] Hoff, T. (2009a). Building Super Scalable Systems: Blade Runner Meets Autonomic Computing in the Ambient Cloud. Disponível online em http://highscalability.com/blog/2009/12/16/buildingsuperscalablesystemsbladerunnermeetsautonomic.html.

[Hoff 2009b] Hoff, T. (2009b). Why are Facebook, Digg, and Twitter so Hard to Scale? Blog post, Disponível online em http://highscalability.com/blog/2009/10/13/why-are-facebook-digg-and-twitter-so-hard-to-scale.html.

[Johnston 2009] Johnston, S. (2009). Introduciong the Cloud Computing Stack. Disponível online em http://samj.net/2009/04/introducing-cloud-computing-stack-2009.html.

[Kandula et al. 2009] Kandula, S., Padhye, J., and Bahl, P. (2009). Flyways To De-Congest Data Center Networks. In *Proc. of workshop on Hot Topics in Networks (HotNets-VIII)*.

[Leavitt 2010] Leavitt, N. (2010). Will NoSQL Databases Live Up to Their Promise? *Computer*, 43:12–14.

[Leighton] Leighton, F. T. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*. Morgan Kaufmann Publishers, 1 edition.

[Lim et al. 2008] Lim, K., Ranganathan, P., Chang, J., Patel, C., Mudge, T., and Reinhardt, S. (2008). Understanding and Designing New Server Architectures for Emerging Warehouse-Computing Environments. In *ISCA '08: Proceedings of the 35th International Symposium on Computer Architecture*, pages 315–326, Washington, DC, USA. IEEE Computer Society.

[McKeown et al. 2008] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., and Turner, J. (2008). OpenFlow: Enabling Innovation in Campus Networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74.

[Mell and Grance 2009] Mell, P. and Grance, T. (2009). The NIST Definition of Cloud Computing. *National Institute of Standards and Technology, Information Technology Laboratory*.

[Microsoft 2010] Microsoft (2010). Windows Azure Platform. Disponível online em http://www.microsoft.com/windowsazure/.

[Mysore et al. 2009] Mysore, R. N., Pamboris, A., Farrington, N., Huang, N., Miri, P., Radhakrishnan, S., Subramanya, V., and Vahdat, A. (2009). PortLand: A Scalable Fault-Tolerant Layer 2 Data Center Network Fabric. *In Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication, Barcelona, Spain*.

[Nanodatacenters 2010] Nanodatacenters (2010). Disponível online em http://www.nanodatacenters.eu.

[NASA 2010] NASA (2010). Nebula. Disponível online em http://www.datacenterknowledge.com/archives/2009/12/02/nasas-nebula-the-cloud-in-a-container/.

[NCSA 2010] NCSA (2010). National Center for Supercomputing Applications. Disponível online em http://www.ncsa.illinois.edu/.

[No-SQL 2010] No-SQL (2010). Disponível online em http://nosql.net.

[Open Cloud Manifesto 2010] Open Cloud Manifesto (2010). Disponível online em http://opencloudmanifesto.org.

[Ousterhout et al. 2009] Ousterhout, J., Agrawal, P., Erickson, D., Kozyrakis, C., Leverich, J., Mazières, D., Mitra, S., Narayanan, A., Parulkar, G., Rosenblum, M., Rumble, S. M., Stratmann, E., and Stutsman, R. (2009). The case for RAMClouds: Scalable High-performance Storage Entirely in DRAM. *SIGOPS Oper. Syst. Rev.*, 43(4):92–105.

[Pujol et al. 2009] Pujol, J. M., Siganos, G., Erramilli, V., and Rodriguez, P. (2009). Scaling Online Social Networks without Pains. NetDB 2009, 5th International Workshop on Networking Meets Databases, co-located with SOSP 2009.

[Qureshi et al. 2009] Qureshi, A., Weber, R., Balakrishnan, H., Guttag, J., and Maggs, B. (2009). Cutting the Electric Bill for Internet-scale Systems. In *SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, pages 123–134, New York, NY, USA. ACM.

[S. Kandula and Patel 2009] S. Kandula, Sudipta Sengupta, A. G. and Patel, P. (2009). The Nature of Data Center Traffic: Measurements and Analysis. In *ACM SIGCOMM Internet Measurement Conference (IMC)*.

[Sonnek and Chandra 2009] Sonnek, J. and Chandra, A. (2009). Virtual Putty: Reshaping the Physical Footprint of Virtual Machines. In *Proc. of Workshop on Hot Topics in Cloud Computing (HotCloud'09)*.

[Tavakoli et al. 2009] Tavakoli, A., Casado, M., Koponen, T., and Shenker, S. (2009). Applying NOX to the Datacenter. In *Proc. of workshop on Hot Topics in Networks (HotNets-VIII)*.

[Valancius et al. 2009] Valancius, V., Mundada, Y., Feamster, N., Rexford, J., and Nakao, A. (2009). Transit Portal: Bringing Connectivity to The Cloud. SIGCOMM 2009 Poster/Demo Session.

[Vaquero et al. 2009] Vaquero, L. M., Rodero-Merino, L., Caceres, J., and Lindner, M. (2009). A Break in the Clouds: Towards a Cloud Definition. *SIGCOMM Comput. Commun. Rev.*, 39(1):50–55.

[Wang et al. 2009] Wang, G., Andersen, D. G., Kaminsky, M., Kozuch, M., Ng, T. S. E., Papagiannaki, K., Glick, M., and Mummert, L. (2009). Your Data Center Is a Router: The Case for Reconfigurable Optical Circuit Switched Paths. In *Proc. of workshop on Hot Topics in Networks (HotNets-VIII)*.

[Wischik et al. 2008] Wischik, D., Handley, M., and Braun, M. B. (2008). The resource pooling principle. *SIGCOMM Comput. Commun. Rev.*, 38(5):47–52.

[Wood et al. 2009] Wood, T., Alexander, Ramakrishnan, K., Shenoy, P., and Merwe, J. V. (2009). The Case for EnterpriseReady Virtual Private Clouds. In *Proc. of Workshop on Hot Topics in Cloud Computing (HotCloud'09)*.

[Wu et al. 2009] Wu, H., Lu, G., Li, D., Guo, C., and Zhang, Y. (2009). MDCube: A High Performance Network Structure for Modular Data Center Interconnection. *In Proceedings of the ACM CONEXT 2009, Rome, Italy*.

[Yuan et al. 2007] Yuan, X., Nienaber, W., Duan, Z., and Melhem, R. (2007). Oblivious Routing for Fat-tree Based System Area Networks with Uncertain Traffic Demands. *SIGMETRICS Perform. Eval. Rev.*, 35(1):337–348.

Capítulo

4

Redes Cognitivas: Um Novo Paradigma para as Comunicações Sem Fio

Marcelo Portela Sousa[1,2], Rafael Fernandes Lopes[1,2,3],
Waslon Terllizzie Araújo Lopes[1,2], Marcelo Sampaio de Alencar[1,2]

[1] Universidade Federal de Campina Grande – UFCG – Campina Grande, Brasil
[2] Instituto de Estudos Avançados em Comunicações – IECOM – Campina Grande, Brasil
[3] Instituto Federal do Maranhão – IFMA – São Luís, Brasil

Abstract

Several areas of human activity depend on the existence of mobile communications services. However, despite the high demand for spectrum, studies indicate its sub-utilization. The limited availability of spectrum and the inefficiency of its usage generate a demand for new mechanisms and paradigms that can exploit the spectrum opportunistically. In this context, a new concept named Cognitive Network *arises, a networking technology that allows efficient utilization of the allocated frequency bands by opportunistic access to these bands. In this chapter the main concepts related to technology and architecture of cognitive networks are presented, in order to create a basic theoretical foundation for the subject.*

Resumo

Diversos setores da atividade humana dependem da existência de serviços de comunicações móveis. No entanto, apesar da grande procura por faixas de espectro, estudos indicam sua subutilização. A limitada disponibilidade de faixas de espectro e a ineficiência de seu uso geram demandas por novos mecanismos e paradigmas que possam explorar o espectro de maneira oportunista. Nesse contexto, surge um conceito denominado Redes Cognitivas, uma tecnologia de rede que possibilita um melhor aproveitamento de faixas de frequência alocadas, porém subutilizadas, por meio do acesso oportunista a estas faixas. Neste capítulo os principais conceitos relacionados à tecnologia e à arquitetura de redes cognitivas são apresentados, com vistas a criar uma fundamentação teórica básica sobre o assunto.

4.1. Introdução

Os serviços de comunicações móveis têm sido usados em diferentes contextos, provendo desde comunicações celulares até o compartilhamento de dados em redes de computadores sem fio. Para evitar interferência entre os sinais de rádio transmitidos, as agências governamentais estabelecem políticas de alocação do espectro de rádio-frequência (RF) [21, 68], que estão geralmente vinculadas ao pagamento de licenças de uso.

A considerável quantidade de serviços de comunicações criados nos últimos anos tem sido responsável pela demanda por alocação do espectro de RF junto às agências de regulamentação, levando à escassez de recursos espectrais em diversas localidades. Com a maior parte do espectro de rádio já alocado, destinar faixas livres para novos serviços ou melhorar os já existentes tem se tornado uma tarefa cada vez mais difícil [30].

Apesar da grande procura por algumas faixas de espectro, estudos indicam uma sub-utilização delas. Uma pesquisa realizada pela Força Tarefa em Políticas de Espectro da Comissão de Comunicações Federal (FCC –*Federal Communications Commission*) aponta uma considerável variação temporal e geográfica no uso do espectro alocado (entre 15 e 85%) [20].

Cabric et al. [8] apresentam em seu trabalho uma medição do uso do espectro no centro da cidade de Berkeley. A Figura 4.1 apresenta o gráfico da densidade espectral de potência obtida na medição[1]. Esse gráfico indica uma baixa utilização do espectro alocado, especialmente nas faixas de 3 a 6 GHz.

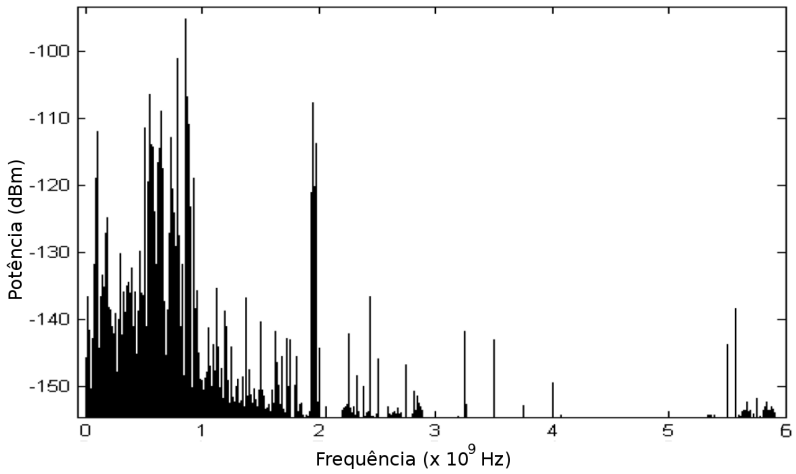


Figura 4.1. Medição da utilização do espectro de 0-6 GHz no centro de Berkeley [8].

A limitada disponibilidade de espectro e a ineficiência de sua utilização geram demandas por novos mecanismos e paradigmas de comunicações que explorem o espectro existente de maneira mais eficaz [1]. As Redes Cognitivas, também denominadas Redes de Rádio Cognitivo ou Redes sem fio de Próxima Geração [2, 50, 73], representam uma tecnologia de rede que aumenta a eficiência da alocação espectral, por meio do acesso oportunista às faixas de frequência.

[1]Os sinais foram coletados em intervalos de tempo de 50 μs, amostrados a uma taxa de 20×10^9 amostras/s

A rede cognitiva foi primeiramente definida por Thomas et al. como [73]:

> *... uma rede dotada de capacidade cognitiva, que pode perceber as condições atuais da rede e então planejar, decidir e atuar sobre essas condições. A rede pode aprender a partir dessas adaptações e utilizar essas informações para tomar futuras decisões, enquanto leva em consideração os objetivos de transmissão fim-a-fim.*

As redes cognitivas fornecem aos usuários móveis uma grande largura de banda por meio do uso de técnicas de acesso dinâmico ao espectro sobre arquiteturas heterogêneas de redes sem fio. Essa nova tecnologia permite uma utilização mais eficiente do espectro, provendo acesso oportunista às faixas licenciadas sem, no entanto, interferir com seus usuários (usuários primários). Entretanto, o desenvolvimento de tecnologias de redes cognitivas impõe desafios de pesquisa, devido à grande faixa espectral a ser gerenciada e aos diversos requisitos de qualidade de serviço (QoS – *Quality of Service*) das aplicações.

De forma a adaptar automaticamente seus enlaces e protocolos de comunicação e utilizar o espectro de maneira oportunista é necessário que as redes cognitivas sejam dotadas de conhecimento contextual derivado de seu ambiente [52, 60]. Por meio da ciência do contexto, os nós da rede podem reconhecer e selecionar automaticamente a tecnologia de acesso sem fio a ser utilizada, em função de sua disponibilidade, localização e tempo [24]. Assim, entre as principais informações de contexto do ambiente, merecem destaque as relacionadas à ocupação espectral.

A principal tecnologia empregada no desenvolvimento da infraestrutura de redes cognitivas são os Rádios Cognitivos (*Cognitive Radios*) [53, 55]. Assim como as redes cognitivas, os rádios cognitivos também fornecem a capacidade de utilizar ou compartilhar o espectro de uma maneira oportunista. No entanto, enquanto os rádios cognitivos atuam apenas nas camadas física e de enlace do modelo de referência ISO/OSI, as redes cognitivas cobrem todas as camadas deste modelo.

Mais especificamente, a tecnologia de rádio cognitivo permite aos seus usuários [2]:

1. Determinar quais faixas do espectro estão disponíveis e detectar a presença de usuários primários, quando há comunicação em uma faixa licenciada (sensoriamento espectral);

2. Selecionar o melhor canal disponível para transmissão (gerenciamento espectral);

3. Compartilhar o acesso a esse canal com outros usuários (compartilhamento espectral);

4. Disponibilizar o canal quando um usuário primário é detectado, mantendo a comunicação enquanto migra para outra faixa (mobilidade espectral ou *handoff* espectral).

Para realizar as funções apresentadas é necessário que os protocolos de comunicação sejam capazes de se adaptar à disponibilidade de faixas de espectro. Dessa forma, os rádios cognitivos utilizam métricas de desempenho das condições atuais de cada camada da pilha de protocolos para determinar a configuração ótima de operação da rede. Para tanto, as redes cognitivas necessitam utilizar uma abordagem de projeto de relacionamento entre camadas (*cross-layer design*), com vistas à obtenção de um desempenho ótimo.

Os componentes de comunicação de uma rede cognitiva e suas interações são ilustradas na Figura 4.2. As funções de sensoriamento e compartilhamento espectral interagem entre si para melhorar a eficiência da alocação de faixas do espectro, interagindo com as camadas física e de enlace. Por sua vez, as funções de gerenciamento e mobilidade espectral atuam sobre todas as camadas do modelo OSI, obtendo informações e mudando suas configurações de acordo com a natureza dinâmica do espectro.

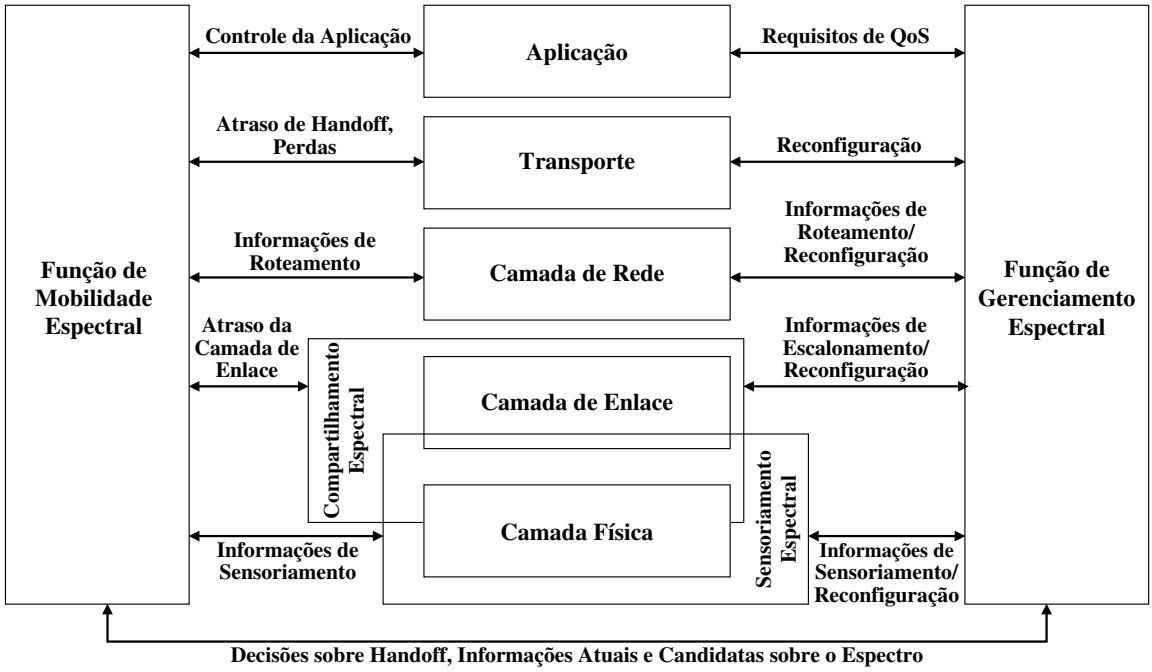


Figura 4.2. Funcionalidades de comunicação de uma rede cognitiva [2].

Os protocolos de comunicação disponíveis atualmente não são adequados para aplicação em redes cognitivas, pois a utilização dinâmica do espectro pode causar efeitos adversos em seu desempenho, principalmente aos protocolos sensíveis à latência (*e.g., streaming* multimídia). Assim, é necessário que sejam desenvolvidos protocolos que, baseados nas informações sobre o estado atual do espectro, possam modificar automaticamente sua tecnologia de transmissão e parâmetros de configuração em função da ocupação espectral. Esse é um importante tópico de pesquisa da área de redes cognitivas.

Neste capítulo, os principais conceitos relacionados à tecnologia e arquitetura de redes cognitivas são apresentados, com vistas a criar uma fundamentação teórica básica sobre o tema. As principais técnicas e algoritmos empregados no processo de descoberta, utilização e compartilhamento do espectro são abordadas, bem como os desafios relati-

vos aos serviços providos por camadas superiores do modelo OSI. Por fim, os principais projetos, tendências e pesquisas em desenvolvimento na área são apresentados.

Este capítulo está organizado da seguinte maneira: a Seção 4.2 apresenta os componentes básicos da arquitetura geral das redes cognitivas, suas funções e cenários de aplicação da tecnologia; a Seção 4.3 aborda aspectos relacionados ao projeto da camada física dos rádios cognitivos; na Seção 4.4 as principais propriedades, vantagens e fatores limitantes dos protocolos de controle de acesso ao meio para redes cognitivas são discutidos; a Seção 4.5 discute algumas características relativas ao processo de roteamento em redes cognitivas; a Seção 4.6 apresenta algumas aplicações de redes cognitivas; a Seção 4.7 aborda o padrão IEEE 802.22 que utiliza a tecnologia de redes cognitivas para transmissões em redes sem fio regionais, por meio do acesso não licenciado ao espectro de TV; por fim, a Seção 4.8 apresenta as considerações finais deste capítulo.

4.2. Arquitetura Geral das Redes Cognitivas

Os protocolos e as tecnologias de redes sem fio existentes apresentam limitações em relação à sua capacidade de adaptação [74]. Essas adaptações são tipicamente reativas, sendo executadas após a ocorrência de algum evento. Além disso, os nós da rede geralmente não realizam trocas de informações sobre seu estado atual. Dessa forma, os nós da rede desconhecem as condições experimentadas por outros elementos, o que inviabiliza a criação de uma visão geral sobre o estado da rede, resultando em comunicações com desempenho sub-ótimo.

Além disso, as arquiteturas de redes sem fio atuais são bastante heterogêneas em termos de políticas de espectro e tecnologias de comunicação [1]. Essa heterogeneidade impõe desafios ao projeto de protocolos para redes cognitivas. Assim, uma completa definição da arquitetura dessas redes é necessária.

A Figura 4.3 apresenta a arquitetura geral das redes cognitivas proposta por Akyildiz et al. [2]. Em um ambiente de redes cognitivas, algumas porções do espectro estão licenciadas para diferentes propósitos, enquanto outras permanecem não licenciadas. Dessa forma, os componentes da arquitetura geral de redes cognitivas podem ser classificados como primários ou licenciados e cognitivos ou não licenciados.

Os elementos básicos das redes primárias e cognitivas são definidos a seguir [2]:

- **Rede Primária** – infraestrutura de rede já existente e que tem direitos de acesso exclusivos a uma certa faixa do espectro (*e.g.,* redes celulares e de televisão). Os componentes de uma rede primária são:
  - Usuário Primário: um usuário primário (ou licenciado) tem licença para operar em uma determinada faixa do espectro. Seu acesso não deve ser afetado por transmissões de usuários não licenciados. Esses usuários não necessitam modificar sua infraestrutura para coexistir com estações radiobase e usuários cognitivos;
  - Estação Radiobase Primária: componente fixo da infraestrutura de rede que tem licença de acesso ao espectro (*e.g.,* transceptor de uma estação radiobase

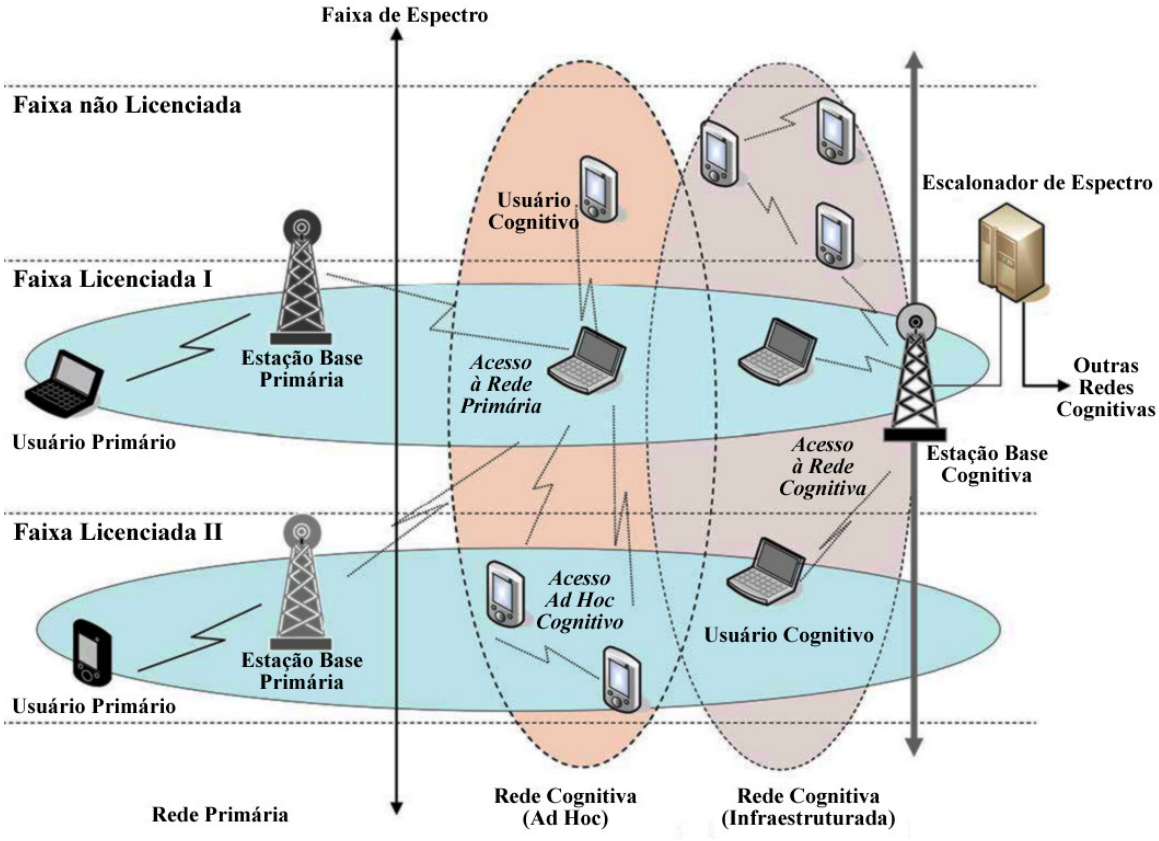


Figura 4.3. Arquitetura de redes cognitivas [2].

em um sistema celular). A estação radiobase primária não apresenta capacidade cognitiva para o compartilhamento do espectro com usuários cognitivos;

- **Rede Cognitiva ou Secundária** – infraestrutura de rede que não tem licença para operar em uma determinada faixa. Seu acesso ao espectro é realizado somente de maneira oportunista. As redes cognitivas podem operar em modo infraestruturado ou *ad hoc*. Os componentes de uma rede cognitiva são:
  - Usuário Cognitivo, Secundário ou não Licenciado: usuário que não tem nenhuma licença de uso do espectro. Esses usuários fazem uso das funcionalidades de compartilhamento de faixas licenciadas do espectro para realizar comunicações;
  - Estação Radiobase Cognitiva, Secundária ou não Licenciada: componente fixo da infraestrutura da rede cognitiva. A estação radiobase cognitiva fornece mecanismos de conexão de salto único aos usuários cognitivos. Por meio das estações base um usuário cognitivo pode ter acesso à rede fixa e a outras redes cognitivas;
  - Escalonador de Espectro: os dispositivos de uma rede cognitiva capturam sua visão local da atividade espectral. Essa informação pode ser diretamente compartilhada entre eles ou agregada em uma base de dados central que provê informações sobre os emissores locais, as políticas de acesso e a área em que os

sinais são transmitidos [24]. O escalonador de espectro é uma entidade central da rede responsável pelo armazenamento dessas informações, além de organizar o acesso aos recursos espectrais entre diferentes redes cognitivas. Ele é um gerenciador de informações de recursos espectrais, e permite a coexistência entre múltiplas redes cognitivas [7, 38].

A arquitetura de referência das redes cognitivas (apresentada na Figura 4.3) consiste de diferentes tipos de redes: (a) uma rede primária, (b) uma rede cognitiva infraestruturada e (c) uma rede cognitiva *ad hoc*. As redes cognitivas são operadas em um ambiente heterogêneo que consiste de faixas licenciadas e não licenciadas. Além disso, os usuários cognitivos podem se comunicar uns com os outros por meio de múltiplos saltos ou estações radiobase.

Nas redes cognitivas existem três diferentes formas de acesso [2]:

- Redes de Acesso Cognitivas – os usuários cognitivos podem acessar as estações base cognitivas em faixas licenciadas ou não licenciadas;
- Redes Cognitivas *Ad hoc* – os usuários cognitivos podem se comunicar com outros usuários cognitivos por meio de conexões *ad hoc* em faixas licenciadas ou não licenciadas;
- Redes de Acesso Primárias – os usuários cognitivos podem ainda acessar as estações radiobase primárias por meio de faixas licenciadas para fazer uso de seus serviços.

De acordo com a arquitetura de referência, diversos mecanismos são necessários para permitir a heterogeneidade em redes cognitivas. A seção a seguir aborda as funcionalidades necessárias à operação das redes cognitivas.

4.2.1. Funcionalidades das Redes Cognitivas

As redes cognitivas podem operar tanto em faixas licenciadas quanto não licenciadas [2]. Consequentemente, as funcionalidades requeridas pelas redes cognitivas variam de acordo com o tipo de faixa acessada. Esta seção apresenta as funcionalidades existentes em redes cognitivas operando em faixas licenciadas e não licenciadas.

4.2.1.1. Operação de Rede Cognitiva em Faixa Licenciada

Conforme discutido na Seção 4.1, existe uma considerável variação na utilização do espectro de radiofrequência licenciado. Consequentemente, as redes cognitivas podem explorar essas faixas do espectro de maneira oportunista, por meio do uso de técnicas de acesso dinâmico ao espectro. Assim, as redes cognitivas devem ter mecanismos que permitam sua coexistência com redes primárias na mesma faixa do espectro.

Os desafios para a operação de redes cognitivas em faixas licenciadas derivam da existência de usuários primários nestas faixas. Ao operar em faixas licenciadas, as

redes cognitivas devem detectar os usuários primários, visto que a capacidade de transmissão dos canais vagos do espectro depende da interferência dos usuários primários próximos [2]. Portanto, evitar interferência com os usuários primários é um dos aspectos mais importantes da arquitetura das redes cognitivas. Além disso, se um usuário primário começar a utilizar uma faixa do espectro alocada por um usuário cognitivo, este deve imediatamente desocupar a faixa atual e migrar para outra faixa disponível (*i.e.*, realizar *handoff* espectral).

4.2.1.2. Operação de Rede Cognitiva em Faixa não Licenciada

A política de abertura de determinadas faixas do espectro, que iniciou com a banda industrial, científica e médica (ISM – *Industrial, Scientific and Medical*), permitiu o desenvolvimento de uma variedade de importantes tecnologias e aplicações inovadoras. Entretanto, a utilização da banda ISM por tecnologias de rede heterogêneas tem reduzido a disponibilidade espectral desta faixa, levando ao aumento de interferências. A capacidade de acesso ao espectro aberto e a qualidade de serviço que essas tecnologias podem oferecer dependem das técnicas de projeto empregadas pelos rádios para alocação eficiente do espectro.

As redes cognitivas podem ser projetadas para operação em faixas não licenciadas, melhorando a eficiência nesta porção do espectro. Dado que não existem proprietários de licenças, todos os nós da rede têm os mesmos direitos de acesso às faixas do espectro. Múltiplas redes cognitivas podem coexistir na mesma área e se comunicar utilizando as mesmas faixas do espectro. Algoritmos de compartilhamento inteligente do espectro podem melhorar a eficiência no uso do espectro e fornecer uma alta qualidade de serviço [2].

Nessa arquitetura, os usuários cognitivos detectam as transmissões de outros usuários cognitivos. Diferentemente das operações em faixas licenciadas, um *handoff* espectral não é provocado pelo aparecimento de outros usuários primários [2]. Entretanto, dado que todos os usuários cognitivos têm os mesmos direitos de acesso ao espectro, eles devem competir entre si pelas mesmas faixas não licenciadas. Assim, nessa arquitetura, métodos de compartilhamento do espectro devem ser empregados pelos usuários cognitivos.

4.3. Projeto da Camada Física

Os avanços na tecnologia de rádio têm permitido o desenvolvimento de técnicas de acesso dinâmico ao espectro eletromagnético e de configuração adaptativa dos enlaces e protocolos de comunicação. A utilização dessas técnicas permite às aplicações se beneficiarem de canais de comunicação com melhor desempenho e menor interferência. Nesse contexto, os rádios cognitivos representam um novo paradigma para as comunicações sem fio, no qual os nós da rede são dotados da capacidade de modificar seus parâmetros de transmissão e recepção, de forma a tornar a comunicação mais eficiente, evitando interferência com usuários licenciados e não licenciados. Por meio da tecnologia de rádio cognitivo é possível ter acesso a comunicação altamente confiável, quando e onde for necessário, e ainda tornar mais eficiente a utilização do espectro de rádio [36].

O principal objetivo da tecnologia de rádio cognitivo é fazer uso da melhor faixa de espectro disponível. Para tanto, os rádios cognitivos utilizam sua capacidade cognitiva

e seus recursos de reconfigurabilidade. Considerando que a maior parte do espectro de RF pode estar alocada, um desafio importante dessa tecnologia é o compartilhamento do espectro licenciado sem interferir com as transmissões dos usuários primários [2].

Segundo Haykin [36], as faixas do espectro de RF podem ser classificadas de acordo com o espectro de potência dos sinais de rádio presentes nelas. Essa classificação é apresentada a seguir:

- Espaços negros (*black spaces*): faixas ocupadas por interferências locais, temporárias e de alta potência;
- Espaços cinzas (*grey spaces*): faixas parcialmente ocupadas por interferências de baixa potência;
- Espaços brancos (*white spaces*): faixas livres de interferências de RF, exceto pelo ruído do ambiente (*e.g.,* ruído térmico, ruído impulsivo).

A tecnologia de rádio cognitivo utiliza espaços em branco (também denominados lacunas de espectro) em faixas licenciadas e não-licenciadas de maneira oportunista para realizar a transmissão de informações [11, 71]. Caso essas faixas passem a ser usadas por um usuário licenciado, o rádio cognitivo deve mudar seu canal de operação para outra lacuna espectral, ou permanecer na mesma faixa, alterando sua potência de transmissão ou esquema de modulação, de forma a evitar interferências. Esse esquema é ilustrado na Figura 4.4.

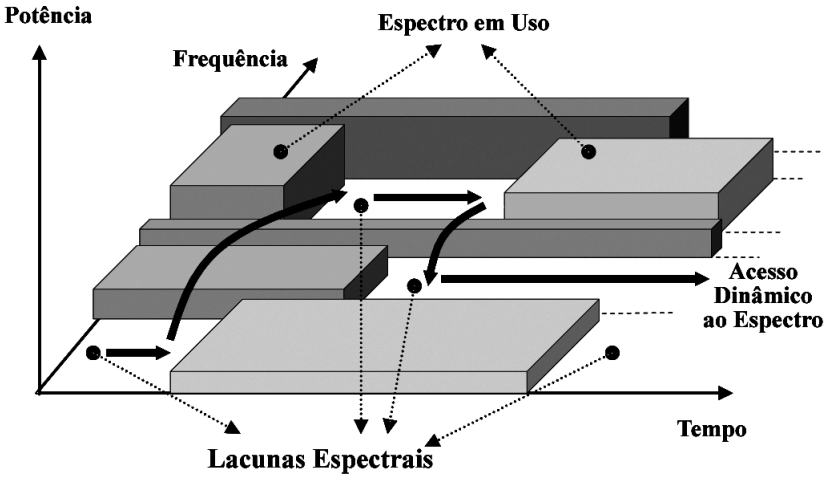


Figura 4.4. Conceito de lacunas espectrais [2].

Considerando que diversos espaços em branco podem estar disponíveis, é necessário que os rádios cognitivos tomem ciência desta diversidade de oportunidades para selecionar o melhor canal disponível [45]. Assim, o próximo desafio é a construção de protocolos de rede que se adaptem dinamicamente à faixa de espectro selecionada [2].

Para permitir uma utilização eficaz do espectro, evitando interferências com usuários primários, é necessário identificar, com confiabilidade, espaços em branco no espectro, em termos de frequência, tempo e espaço [33]. Entre as principais abordagens

empregadas com essa finalidade merecem destaque: registro em banco de dados, sinalizadores regionais e sensoriamento espectral. A Tabela 4.1 apresenta um resumo das características dessas técnicas.

Tabela 4.1. Classificação dos métodos de identificação de espaços em branco [30, 33].

	Custo	**Compatível com sistemas legados**	**Complexidade**	**Posicionamento**	**Conexão à Internet**	**Monitoramento contínuo**	**Canal padronizado**
Registro em banco de dados	Alto		Baixa	X	X		
Sinalizadores regionais	Alto		Baixa	X			X
Sensoriamento espectral	Baixo	X	Alta			X	

Os dois primeiros métodos deixam a cargo dos sistemas primários a tarefa de fornecer aos usuários secundários as informações relativas à utilização atual do espectro [6]. Na primeira abordagem, os usuários primários registram os dados relevantes (*e.g.,* sua localização, potência e tempo de utilização esperado) em um banco de dados centralizado. Os sistemas secundários devem se conectar a esse banco de dados (*e.g.,* por meio da Internet) para determinar a disponibilidade de lacunas espectrais em suas localizações. Alternativamente, as informações de ocupação do espectro fornecidas pelos usuários primários em operação em cada região podem ser difundidas sobre uma determinada área utilizando sinalizadores regionais (*beacons*), dessa forma eliminando a necessidade de uma conexão ao banco de dados. A partir dessas sinalizações, os usuários secundários podem identificar os lacunas de espectro existentes em sua vizinhança.

Apesar de demandarem transceptores secundários mais simples, os métodos apresentados requerem modificações nos atuais sistemas licenciados e, dessa forma, são incompatíveis com sistemas primários legados. Além disso, sua implantação apresenta um alto custo e requer a existência de mecanismos de obtenção de informações de posicionamento aos usuários secundários (além de uma conexão a um banco de dados ou a um canal dedicado e padronizado com os sinalizadores regionais). Por outro lado, o sensoriamento espectral confia apenas na capacidade do sistema secundário de identificar os espaços em branco, com um sensoriamento direto nas faixas licenciadas. Assim, o sistema secundário monitora as faixas de frequência licenciadas e, de maneira oportunista, transmite quando não detecta nenhum sinal primário.

Por conta de seu baixo custo e sua compatibilidade com sistemas primários legados, o sensoriamento espectral tem recebido mais atenção da comunidade científica que as outras abordagens [33]. Essa técnica tem sido a principal alternativa considerada para inclusão em padrões que utilizam a tecnologia de rádio cognitivo, como o padrão IEEE 802.22 [37, 16]. Por outro lado, uma desvantagem dessa abordagem é que os dados de utilização do espectro dos sistemas primários não estão disponíveis *a priori*. Além disso, os usuários secundários devem sensoriar continuamente as faixas licenciadas enquanto a utilizam, de forma a perceber o retorno dos usuários primários às faixas licenciadas. A subseção a seguir apresenta detalhadamente a técnica de sensoriamento espectral.

4.3.1. Sensoriamento Espectral

A modificação dos parâmetros de transmissão, realizada pelos rádios cognitivos, é baseada no monitoramento ativo de diversos fatores externos e internos ao ambiente de rádio, como a ocupação do espectro de RF, o comportamento do usuário e o estado da rede. Esses e outros fatores compõem o conhecimento contextual do ambiente de rádio.

Para manter sua ciência sobre a ocupação do espectro de RF, os rádios cognitivos necessitam verificar frequentemente os canais disponíveis em um amplo espectro. No entanto, esse processo nem sempre resulta em estimativas confiáveis, uma vez que ele se baseia na observação local de sinais cuja potência recebida pode ser baixa, ou mesmo não detectável. Erros nas estimações espectrais podem levar à ocorrência de interferências entre as transmissões. Além disso, durante o sensoriamento espectral, a transmissão de dados pelas aplicações não é possível, resultando em atrasos adicionais e em uma redução na disponibilidade de largura de banda para o tráfego das aplicações [45].

Um rádio cognitivo tem a capacidade de verificar o ambiente espectral sobre uma ampla faixa e explorar esta informação para, oportunisticamente, prover enlaces sem fio que melhor atendam aos requisitos de comunicação dos usuários e aplicações. Esses dispositivos são projetados para serem cientes e sensíveis às mudanças no ambiente ao seu redor, realizando adaptações à medida que detectam espaços em branco. Em geral, a capacidade de sensoriamento espectral está associada às camadas física (PHY) e de controle de acesso ao meio (MAC), conforme ilustrado na Figura 4.5.

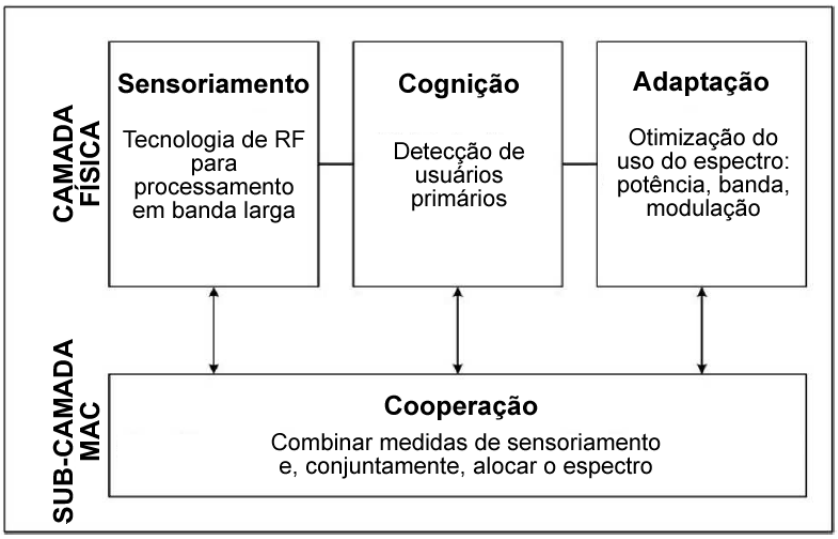


Figura 4.5. Funcionalidades das camadas relacionadas ao sensoriamento espectral [8].

Dado que os rádios cognitivos são considerados usuários de menor prioridade do espectro licenciado, um requisito fundamental é evitar interferências com potenciais usuários primários em sua vizinhança. Por outro lado, sistemas primários não precisam modificar sua infraestrutura para o compartilhamento do espectro com redes cognitivas. Os rádios cognitivos devem ser capazes de detectar a presença de usuários primários por meio de um processo contínuo de sensoriamento.

Classes diferentes de usuários primários podem requerer níveis de sensibilidade e

taxas de sensoriamento distintos para sua detecção. Por exemplo, sinais de difusão de TV são mais facilmente detectados que sinais de um sistema de posicionamento global (GPS – *Global Positioning System*), dado que a sensibilidade dos receptores de TV é dezenas de decibéis menor que a dos receptores de GPS [8].

Em geral, a sensibilidade dos rádios cognitivos deve superar a dos receptores dos usuários primários por uma ampla margem. Essa margem é necessária porque o rádio cognitivo não pode obter uma medição direta do canal entre o receptor e o transmissor primário, e deve basear sua decisão na medição local dos sinais emitidos pelo transmissor primário. Esse tipo de detecção é chamada de sensoriamento espectral local, e pode sofrer com o problema de ocultação de terminais, que pode ocorrer quando o rádio cognitivo está sombreado, sofrendo um severo desvanecimento por multipercurso, ou localizado dentro de construções com alta perda por penetração [8]. Uma possível abordagem para o tratamento desse problema pode ser a adoção de técnicas de sensoriamento espectral colaborativo, em que diversos rádios compartilham suas informações sobre a ocupação do espectro e realizam, conjuntamente, a detecção de sinais primários [32].

4.3.1.1. Arquitetura Física para o Sensoriamento Espectral

Para prover a capacidade de modificar seus parâmetros de operação dinamicamente, a infraestrutura dos rádios cognitivos utiliza a tecnologia de rádio definido por *software* (SDR – *Software Defined Radio*). Os SDRs são sistemas de comunicação de rádio em que os componentes, tipicamente implementados em *hardware* (*e.g., mixers*, filtros, amplificadores, moduladores/demoduladores, detectores, etc.), são implementados em *software*, criando grande flexibilidade em sua operação [23, 54]. A Figura 4.6 apresenta a arquitetura física geral dos transceptores SDR.

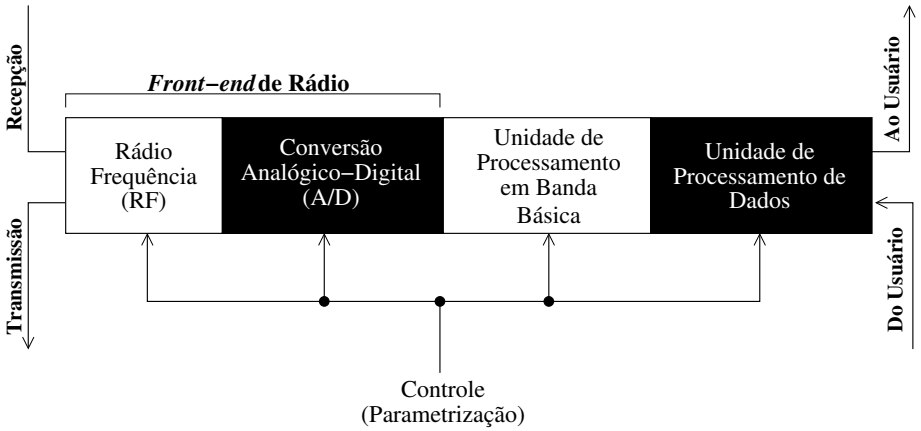


Figura 4.6. Arquitetura física geral dos SDRs [41].

Os principais componentes da arquitetura física dos SDRs são o *front-end* de rádio, que permite a recepção de sinais em um amplo espectro de frequências e a sua digitalização, e a unidade de processamento em banda básica, responsável por realizar o processamento dos sinais e, posteriormente, dos dados recebidos. Os componentes dessa arquitetura podem ser reconfigurados por meio de um barramento de controle que fornece

às unidades de processamento diversos parâmetros de configuração. Essa infraestrutura, chamada de SDR controlado por parâmetros (*parameter-controlled SDR* ou *PaC SDR*), garante que as configurações da transmissão possam ser modificadas instantaneamente, caso necessário [41].

4.3.2. Técnicas de Sensoriamento Espectral

O rádio cognitivo deve distinguir faixas do espectro livres e ocupadas. Para tanto, ele deve ter a capacidade de determinar se o sinal de um transmissor primário está presente em uma certa faixa do espectro.

Nessa abordagem, o rádio cognitivo realiza o monitoramento de frequências licenciadas, por meio de observações locais. Quando sinais primários não são detectados em uma determinada faixa, o rádio cognitivo passa a utilizar o canal de forma oportunista. Apesar de parecer similar à abordagem *listen-before-talk* (ouvir antes de falar) de sensoriamento de portadora física (empregado nas redes IEEE 802.11), existem efetivamente algumas diferenças importantes entre os dois, decorrentes dos rigorosos requisitos de não-interferência impostos para proteger os sistemas primários [30]. Por exemplo, ao alocar uma faixa do espectro, um rádio cognitivo deve continuar o monitoramento do canal e, caso um sinal licenciado seja detectado, ele deve disponibilizar o canal [8, 32].

O modelo básico de hipótese para a detecção de transmissores pode ser definido a seguir [32]

$$x(t) = \begin{cases} n(t) & : \ H_0, \\ hs(t) + n(t) & : \ H_1 \end{cases} \tag{1}$$

em que $x(t)$ é o sinal recebido pelo usuário cognitivo, $s(t)$ é o sinal transmitido pelo usuário primário, $n(t)$ é o ruído aditivo do canal e h é o ganho de amplitude do canal. H_0 é a hipótese nula, em que que não existe nenhum sinal primário em uma certa faixa do espectro. Por outro lado, H_1 é a hipótese alternativa, que indica que existe um sinal de usuário primário.

Baseado no conhecimento dos sistemas secundários sobre a estrutura de sinais primários e suas características, diferentes métodos de sensoriamento podem ser utilizados para distinguir os espaços em branco de faixas ocupadas. Esses métodos podem ser classificados em três tipos [30]:

(a) Filtragem Casada;

(b) Detecção de Energia;

(c) Detecção de Características Cicloestacionárias.

As subseções a seguir apresentam os métodos de sensoriamento espectral.

4.3.2.1. Filtragem Casada

A forma ótima para a detecção de sinais em ruído estacionário gaussiano é a utilização de filtros casados [61]. Entretanto, um filtro casado realiza a demodulação efetiva do sinal do usuário primário. Isso significa que o rádio cognitivo deve ter conhecimento *a priori* de características do sinal do usuário primário, das camadas física e MAC (*e.g.,* tipo de modulação, formato de pulso, formato de pacote). Para tanto, as informações relativas aos sinais a serem detectados precisam estar pré-armazenadas na memória do rádio cognitivo. Caso essas informações não sejam suficientemente precisas, o filtro casado poderá apresentar um baixo desempenho [2].

Para detectar o sinal do usuário primário é necessário realizar a sincronização temporal e de portadora. Além disso, dado que muitos sistemas de rede sem fio apresentam portadoras piloto, preâmbulos, palavras de sincronização ou códigos de espalhamento, estes podem ser usados para a detecção coerente.

A principal vantagem da utilização de filtros casados para sensoriamento espectral é que ele requer menos tempo de observação para atingir um certo nível de sensibilidade [70]. Por outro lado, é necessário que o sinal primário seja demodulado pela unidade de sensoriamento. Assim, os rádios cognitivos devem implementar todos os métodos de detecção relativos aos usuários primários que poderão ser detectados, aumentando a complexidade da unidade de sensoriamento. Essa abordagem é viável apenas no caso do sistema secundário operar em algumas poucas faixas primárias, como é o caso do padrão IEEE 802.22, que se propõe a utilizar faixas de TV de forma oportunista para comunicação em redes regionais (WRAN – *Wireless Regional Area Network*) [37, 16]. O custo de implementação e a complexidade associadas a essa abordagem aumenta à medida que mais faixas primárias são utilizadas de forma oportunista. Outra desvantagem associada a essa abordagem é o alto consumo de energia, uma vez que é necessário executar vários algoritmos de detecção.

4.3.2.2. Detecção de Energia

Caso o receptor não possa obter informações suficientes sobre os sinais dos usuários primários, uma alternativa simples para detectar um sinal primário com ruído é a detecção de energia. Um detector de energia simplesmente mede a energia recebida em uma faixa primária durante um intervalo de observação e a identifica como uma lacuna espectral caso a energia medida seja menor que um limiar apropriadamente definido.

Em níveis baixos de razão sinal-ruído, quando comparado à utilização de filtros casados, a detecção de energia requer um maior tempo de sensoriamento para atingir um bom desempenho [70]. Entretanto, seu baixo custo de implementação e simplicidade tornam essa abordagem um candidato favorável para o sensoriamento espectral em redes cognitivas.

Para realizar a medição de energia de um sinal recebido, o sinal de saída de um filtro passa baixa com largura de banda W é elevado ao quadrado e integrado durante o período de observação T. Finalmente, a saída do integrador Y é comparada a um limiar λ para decidir quando um usuário licenciado está presente (hipótese H_0) ou não (hipótese

H_1) [19]. Esse processo é ilustrado no diagrama de blocos da Figura 4.7.

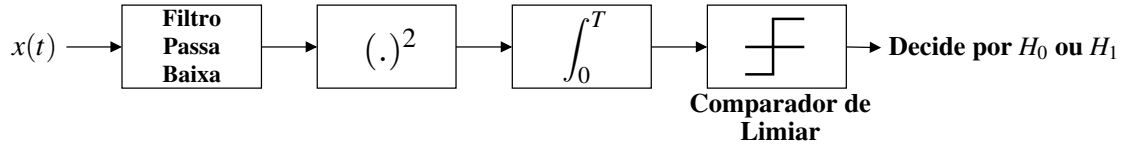


Figura 4.7. Diagrama de blocos de um detector de energia.

Se a detecção de energia puder ser aplicada em um ambiente sem desvanecimento, em que h é o ganho de amplitude do canal (conforme a Equação 1), a probabilidade de detecção P_d e de falso alarme P_f são dadas por [19]

$$P_d = P\{Y > \lambda | H_1\} = Q_m\left(\sqrt{2\gamma}, \sqrt{\lambda}\right), \tag{2}$$

$$P_f = P\{Y > \lambda | H_0\} = \frac{\Gamma(u, \lambda/2)}{\Gamma(u)} \tag{3}$$

em que γ é a relação sinal-ruído, $u = TW$ é o produto tempo-largura de banda, $\Gamma(.)$ e $\Gamma(.,.)$ são as funções gamma completa e incompleta e $Q_m(.,.)$ é a função Marcum-Q generalizada. Assim, enquanto uma baixa P_d pode resultar em estimativas erradas sobre a presença de usuários primários (levando a um baixo aproveitamento do espectro), uma alta probabilidade pode, por sua vez, aumentar a interferência aos usuários primários. Além disso, uma alta P_f pode resultar em uma baixa utilização do espectro, dado que falsos alarmes podem aumentar o número de oportunidades perdidas. Dada sua facilidade de implementação, diversos trabalhos têm adotado a abordagem de detecção de energia [31, 62].

O desempenho do detector de energia é suscetível à incerteza quando ele está imerso em ruído de alta potência. De forma a resolver esse problema, um tom piloto do transmissor primário pode ser utilizado para melhorar a precisão do detector de energia [62].

4.3.2.3. Detecção de Características Cicloestacionárias

A principal desvantagem do detector de energia é a sua falta de habilidade em distinguir entre fontes de energia recebida (sinal primário e ruído), tornando-o suscetível às incertezas relativas à potência do ruído de fundo, especialmente sob baixa razão sinal-ruído [66]. Assim, se algumas características do sinal primário como frequência da portadora ou tipo de modulação forem conhecidos, detectores de características podem ser empregados para lidar com essas informações, ao custo de uma maior complexidade [8, 22, 34, 58].

A análise de sinais aleatórios estacionários é baseada na função de autocorrelação e na densidade espectral de potência. Por outro lado, sinais cicloestacionários exibem correlação entre componentes espectrais separados devido à redundância causada pela periodicidade [26].

Sinais modulados são em geral acoplados com portadoras senoidais, trens de pulsos, sequências de saltos ou prefixos cíclicos, que resultam em uma periodicidade embutida. Esses sinais são caracterizados pela cicloestacionariedade, dado que sua média e autocorrelação exibem periodicidade.

Em seu trabalho, Freitas et al. [25] apresentam alguns exemplos de classificadores de modulação digital utilizados na literatura para o reconhecimento dos padrões de cicloestacionariedade. Os autores também apresentam o algoritmo proposto por [77] para a extração de características cicloestacionárias.

Assim, um sinal $x(t)$ é definido como cicloestacionário de segunda ordem (no sentido amplo) se sua função de autocorrelação

$$R_x(t,\tau) = E[x(t+\tau/2)x(t-\tau/2)] \tag{4}$$

for periódica no tempo t para cada intervalo de tempo τ. Além disso, a função de autocorrelação cíclica (FAC) [26] pode ser utilizada para estudar a cicloestacionariedade de um sinal, e é definida como

$$R_x^{\alpha}(\tau) = \lim_{\Delta t\to\infty} \frac{1}{\Delta t} \int_{-\Delta t/2}^{\Delta t/2} x(t+\tau/2)x(t-\tau/2)e^{-i2\pi\alpha t}dt, \tag{5}$$

em que α é a frequência cíclica (variando entre todos os múltiplos da frequência fundamental) e Δt o intervalo de tempo.

A cicloestacionariedade de segunda ordem especifica o padrão de correlação que ocorre no espectro do sinal. Esse padrão pode ser usado, equivalentemente, para examinar a cicloestacionariedade do sinal e pode ser analisado usando a função de correlação espectral (FCE) [26]

$$S_x^{\alpha}(f) = \lim_{\Delta f\to\infty}\lim_{\Delta t\to\infty} \frac{1}{\Delta t} \int_{-\Delta t/2}^{\Delta t/2} \Delta f X_{1/\Delta f}\left(t, f+\frac{\alpha}{2}\right) \cdot X^*_{1/\Delta f}\left(t, f-\frac{\alpha}{2}\right) dt,$$

em que

$$X_{1/\Delta f}(t,v) = \int_{t-1/2\Delta f}^{t+1/2\Delta f} x(u)e^{-i2\pi vu}du$$

representa a envoltória complexa da componente de faixa estreita do sinal $x(t)$, com frequência central v e largura de faixa Δf. A cicloestacionariedade de segunda ordem de um sinal pode ser examinada por meio da FAC e FCE dos mesmos.

A função de correlação espectral é também chamada de *espectro cíclico*. Diferentemente da densidade espectral de potência, que é um transformação unidimensional de valor real, a FCE é uma transformação bidimensional, geralmente de valor complexo e de parâmetro α (frequência cíclica). A densidade espectral de potência é um caso especial da FCE para $\alpha = 0$. O diagrama de blocos da Figura 4.8 apresenta o esquema geral de implementação do detector de características cicloestacionárias.

A partir da análise espectral do sinal é possível detectar diversas características como o número de sinais, seus tipos de modulação, taxas de símbolo e presença de inter-

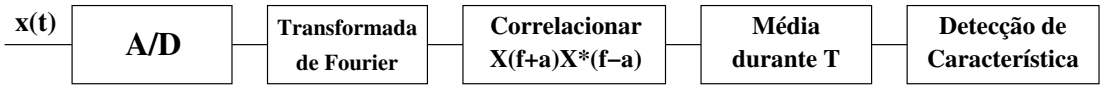


Figura 4.8. Diagrama de blocos da implementação de detectores de características cicloestacionárias.

ferências [8]. Diversas técnicas de reconhecimento de padrões podem ser aplicadas sobre a função de correlação espectral de sinais cicloestacionários como redes neurais, *naive bayes*, máquina de suporte vetorial (SVM – *Support Vector Machine*), *k* vizinhos mais próximos (KNN – *k-nearest neighbor*) e árvores de decisão.

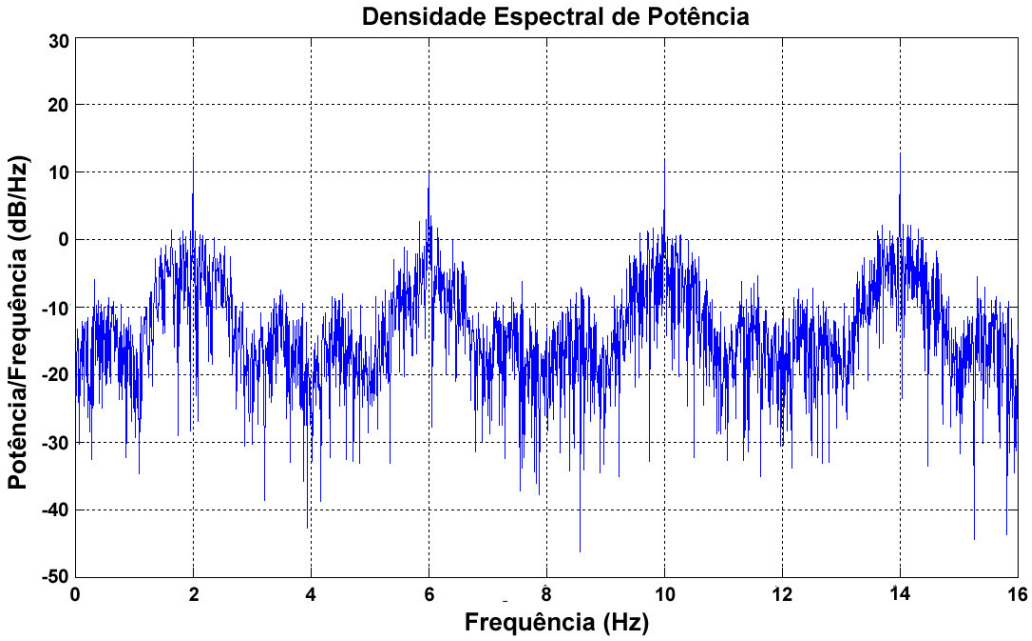
A Figura 4.9 apresenta a densidade espectral de potência e a função de correlação espectral de um sinal com modulação 4-FSK (*Frequency Shift Keying*). Enquanto a primeira medida é utilizada pela técnica de detecção de energia, a segunda é utilizada para a detecção de características cicloestacionárias em meio a ruído.

A principal vantagem da função de correlação espectral é que ela diferencia a energia do ruído da energia do sinal modulado. Isso resulta do fato de que o ruído é um sinal estacionário em sentido amplo e descorrelacionado, enquanto os sinais modulados são cicloestacionários com correlação espectral devido à redundância embutida na periodicidade do sinal [2]. Além disso, um detector de características cicloestacionárias apresenta um melhor desempenho que um detector de energia para diferenciar sinais e o ruído, principalmente por conta de sua robustez em lidar com a incerteza sobre potência do ruído [72]. Entretanto, é computacionalmente complexo e requer um tempo de observação significativamente longo.

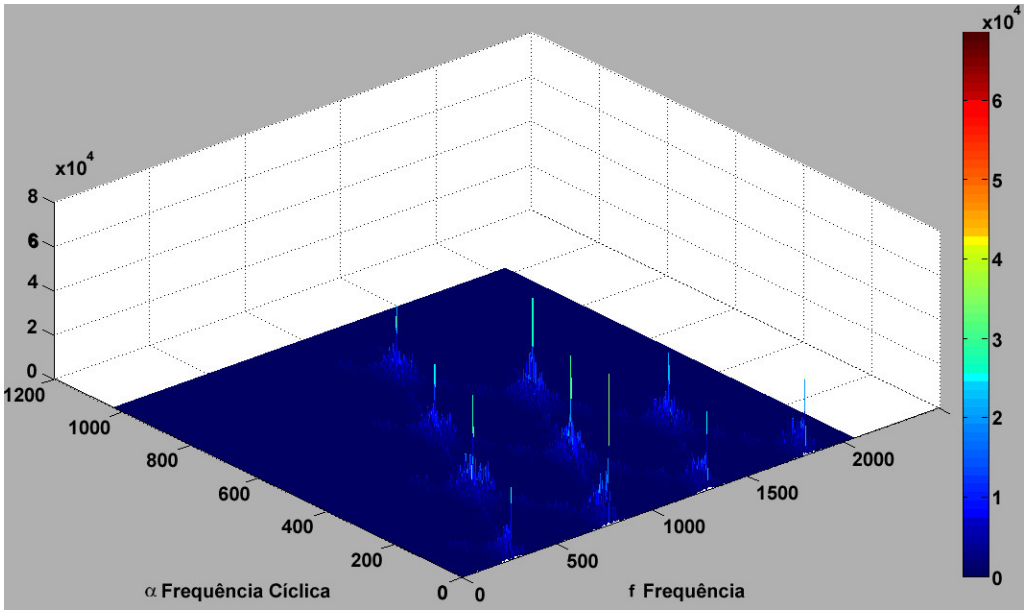
A Tabela 4.2 apresenta um quadro comparativo entre as principais classes de técnicas que podem ser empregadas para o processamento de sinais em sensoriamento espectral. Cada uma dessas técnicas apresenta vantagens e desvantagens, sendo cada uma adequada a diferentes situações. Na prática, uma combinação de diferentes técnicas podem ser utilizadas para tratar diferentes situações.

Tabela 4.2. Comparação das principais classes de técnicas empregadas no sensoriamento espectral [30].

Técnicas de Sensoriamento	Vantagens	Desvantagens
Filtro Casado	• Desempenho Ótimo	• Dependente do sinal • Alto consumo de energia
Detecção de Energia	• Baixa Complexidade • Independe do Sinal Primário	• Suscetível à Incerteza do Ruído • Suscetível à Interferência
Detecção de Características	• Robusto contra Incerteza do Ruído • Robusto contra Interferência	• Computacionalmente Complexo



(a) Densidade espectral de potência do sinal 4-FSK.



(b) Função de correlação espectral do sinal 4-FSK.

Figura 4.9. Densidade espectral de potência e função de correlação espectral de um sinal 4-FSK.

4.4. Controle de Acesso ao Meio (MAC)

Em redes cognitivas, identificar os recursos espectrais disponíveis por meio de técnicas de sensoriamento, decidir os períodos ótimos para transmissão e a coordenação do acesso espectral com outros usuários são funções importantes para os protocolos de acesso ao meio (MAC – *Medium Access Control*). Nesta seção, as principais propriedades, vantagens e fatores limitantes dos protocolos de controle de acesso ao meio para redes

cognitivas são apresentados, no contexto de redes infraestruturadas e *ad hoc*. Além disso, as técnicas de acesso ao espectro em redes cognitivas, como os protocolos de acesso aleatório, acesso agendado e acesso híbrido são abordadas. Os principais desafios e linhas de pesquisa são apresentados, destacando a relação próxima entre os protocolos MAC com o gerenciamento espectral e outras camadas da pilha de protocolos [18].

4.4.1. Gerenciamento Espectral em Redes Cognitivas

As faixas espectrais ociosas detectadas pela fase de sensoriamento espectral apresentam características diferentes, tanto pela natureza variante do ambiente de rádio, quanto por parâmetros da faixa espectral, como frequência de operação e largura de banda. Portanto, as redes cognitivas devem decidir pela melhor faixa espectral de modo a atender os requisitos de QoS, novas funções de gerenciamento de espectro são necessárias. Essas funções são classificadas como sensoriamento espectral, análise espectral e decisão espectral. Enquanto o sensoriamento espectral (descrito na Seção 4.3) está relacionado principalmente com a camada física, a análise e decisão espectral estão relacionadas a camadas superiores da pilha de protocolos [2].

4.4.1.1. Análise Espectral

Os espaços em branco apresentam características diferentes que variam no tempo. A análise espectral habilita a caracterização de diferentes faixas, que podem ser exploradas para a verificação da faixa espectral apropriada, com o intuito de atender os requisitos do usuário. Dessa forma, é essencial definir parâmetros como nível de interferência, taxa de erro do canal, atenuação por percurso e tempo de espera, que podem representar a qualidade de uma faixa de espectro particular [2, 50].

4.4.1.2. Decisão Espectral

Quando todas as faixas do espectro estiverem caracterizadas, operações apropriadas devem ser estabelecidas para a transmissão correspondente, considerando os requisitos de QoS e as características do espectro. Então, a função de gerenciamento do espectro precisa estar ciente dos requisitos de QoS do usuário. Parâmetros tais como taxa de dados, taxa aceitável de erros, limitante de atraso, modo de transmissão, e largura de banda da transmissão podem ser determinados. De acordo com as regras de decisão, o conjunto de faixas espectrais apropriadas pode ser escolhido.

Após selecionar os recursos espectrais por meio das funções de gerenciamento do espectro, o esquema de acesso espectral adequado precisa ser executado. Essa é a principal função do protocolo MAC em redes cognitivas [2].

4.4.2. Acesso ao Meio

O projeto de protocolos MAC para redes cognitivas tem seguido duas propostas distintas [18]. A primeira abordagem está focada principalmente em redes infraestruturadas, em que um coordenador central ou uma estação radiobase gerencia a alocação espectral e compartilha a informação de alocação espectral com os usuários secundários.

Eles, entretanto, podem participar da função de sensoriamento espectral e disponibilizam informações do canal para o controlador central. Os esforços pela padronização levam a uma uniformidade, de modo a permitir que múltiplos operadores de rádio cognitivo coexistam de forma independente.

Por outro lado, a segunda abordagem é otimizada para um tipo particular de ambiente, ou para um objetivo de aplicação específica do usuário. Essa abordagem tem sido bastante aplicada em protocolos distribuídos, que operam sem o suporte de uma entidade de controle centralizada. Como um exemplo, os nós em uma rede *ad hoc* podem exibir elevados graus de mobilidade, o que dificulta a coordenação do sensoriamento. Para tais casos, o protocolo MAC pode identificar a mobilidade com o objetivo de determinar quais regiões (cobertas pelo nó durante seu movimento) exibem altos níveis de atividade de usuários primários [18].

Tanto para redes infraestruturadas, quanto para redes *ad hoc*, os protocolos MAC são classificados em três categorias: acesso aleatório, acesso agendado e acesso híbrido.

4.4.2.1. Protocolos de Acesso Aleatório

Os protocolos de acesso aleatório não necessitam de sincronização temporal e são geralmente baseados no princípio de detecção de portadora de múltiplo acesso com prevenção de colisão (CSMA/CA – *Collision Sense Multiple Access with Collision Avoidance*). O usuário secundário monitora as faixas espectrais para detectar quando não há transmissões provenientes de outros usuários secundários e transmite após um determinado intervalo, para prevenir transmissões simultâneas [18].

Uma estratégia para a utilização de protocolos de acesso aleatório em redes cognitivas infraestruturadas foi proposta em [49]. Esse protocolo garante a coexistência entre os usuários cognitivos e os usuários primários pela adaptação da potência de transmissão e taxa de transmissão da rede cognitiva. Nessa estratégia, as estações radiobase cognitivas e primárias são separadas, já que elas podem apresentar sobreposição das áreas de cobertura. Os usuários cognitivos e primários estabelecem conexões diretas de transmissões de saltos únicos com suas respectivas estações radiobase [49].

O protocolo permite transmissões simultâneas dos usuários cognitivos mesmo quando os usuários primários são detectados, pois a interferência causada a eles está contida em um limiar pré-definido. A rede primária segue um protocolo CSMA, em que os usuários primários realizam o sensoriamento de portadora por um período τ_p antes de transmitir um pacote de requisição de envio (RTS – *Request To Send*) para a estação radiobase correspondente. Se a estação radiobase primária estiver disponível para a transmissão, ela pode responder com uma confirmação para o envio (CTS – *Clear To Send*). Entretanto, o tempo de sensoriamento de portadora realizado pelos usuários cognitivos é maior ($\tau_s >> \tau_p$), de modo que a prioridade do acesso espectral é atribuida aos usuários primários. Baseada na distância dos usuários cognitivos a partir da estação radiobase e na potência do ruído, a estação radiobase decide os parâmetros de transmissão, tais como a potência e taxa de transmissão, para a transferência de dados correspondente. O usuário cognitivo tem a permissão de enviar apenas um pacote em uma rodada dessa negociação, com o objetivo de minimizar o risco de interferência para os usuários primários [49]. A

Figura 4.10 mostra o comportamento detalhado do protocolo em quatro casos diferentes (a-d):

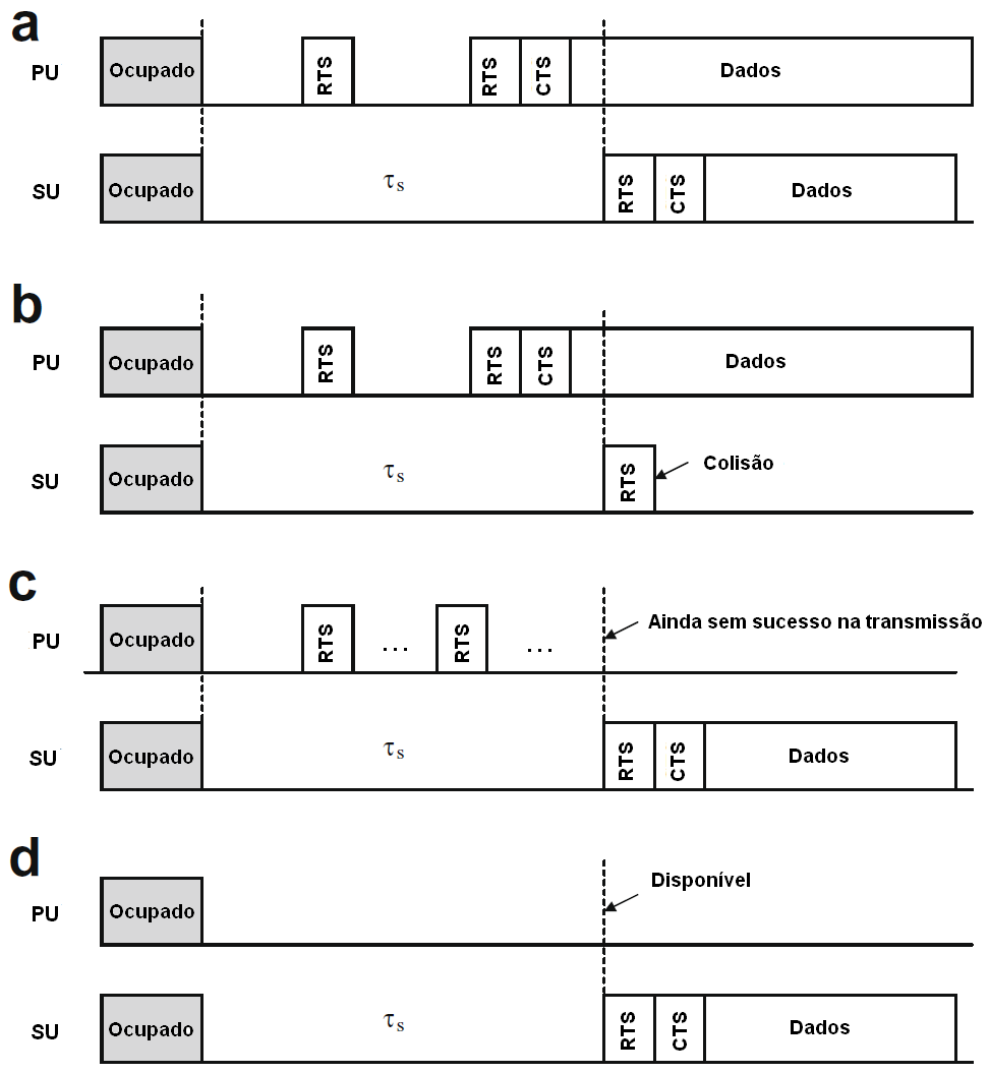


Figura 4.10. Protocolo baseado em CSMA com quatro opções de *handshaking* [18].

Caso a – O usuário primário ganha o acesso ao canal após o sensoriamento de portadora e envia seus dados. O usuário secundário avalia o canal por um período τ_s e ao detectar uma oportunidade de utilização do canal, disputa o acesso ao meio utilizando a sinalização RTS-CTS. Então, esse usuário transmite os dados com potência e taxa sugeridas pela estação radiobase, de modo que transmissões paralelas de usuários primários não sejam afetadas;

Caso b – Nesse caso o pacote RTS enviado pelo usuário secundário está sujeito a colisões. O usuário precisa aguardar pela próxima oportunidade de transmissão depois de repetir o processo de sensoriamento anterior;

Caso c – O usuário primário envia pacotes repetidos de RTS, mas incorre de colisões em

cada período. O usuário secundário pode iniciar a transmissão independentemente da rede primária, *i.e.*, sem ajustar sua potência e taxa de transmissão;

Caso d – O usuário primário não possui pacotes para enviar, de modo que o canal fica disponível durante o período de sensoriamento pel usuário secundário. Similar ao caso anterior, o usuário secundário pode iniciar a transmissão sem considerar a rede primária [18].

Um protocolo MAC de acesso aleatório para redes *ad hoc* foi proposto em [39] para o sensoriamento espectral eficiente e acesso espectral considerando restrições de *hardware*, tais como as limitações operacionais de um terminal de rádio, o sensoriamento espectral parcial e os limites de agregação espectral. Ele usa um canal de controle comum (CCC – *Common Control Channel*), mas também possui um único rádio que simplifica os requisitos de *hardware*. As restrições de *hardware* podem ser divididas em duas classes: restrições de sensoriamento e restrições de transmissão. As restrições de sensoriamento lidam com a relação de compromisso entre o período usado para o sensoriamento e a precisão resultante. Um exemplo é o sensoriamento fino, em que um intervalo de tempo considerável precisa ser alocado por canal e uma porção limitada do espectro é observada. As restrições de transmissão estão relacionadas às limitações impostas pelo esquema de multiplexação por divisão em frequência ortogonal (OFDM – *Orthogonal Frequency Division Multiplexing*) que decide o intervalo da largura de banda, assim como o número máximo permitido de subportadoras. As contribuições do protocolo MAC são [39]:

- **Decisão de Sensoriamento** – De modo a determinar quantos canais devem ser avaliados, um critério de parada para o sensoriamento de sucessivos canais precisa ser decidido. A escolha de um número maior de canais aumenta a largura de banda disponível, levando a uma maior taxa de transmissão de dados. Entretanto, o custo de sensoriamento, especialmente se o canal identificado estiver indisponível, também deve ser considerado. O critério de parada escolhe um intervalo de tempo para interromper a busca de canais, de modo que a recompensa é maximizada. A escolha de quantos canais devem ser avaliados é também determinada pela largura de banda máxima permitida que pode ser acessada pelo transceptor em um dado instante e também pelo número máximo permitido de subportadoras que podem ser usadas nos canais disponíveis nesse intervalo. Os autores de [39] propõem indução reversa para solucionar esse problema e técnicas de redução de tempo computacional, especialmente se o número de canais for grande.

- **Operação do Protocolo** – O protocolo MAC é constituído pelas operações de contenção, sensoriamento e transmissão. Na fase de contenção, os pacotes de C-RTS e o C-CTS enviados sobre o CCC são usados para ganhar acesso ao canal. O terminal transmissor e seu respectivo receptor, que ganham acesso à contenção, trocam pacotes S-RTS e S-CTS para cada canal que é avaliado. Ao final de cada rodada de sensoriamento, a decisão é feita considerando a possibilidade de iniciar o sensoriamento em um um novo canal, baseado no critério de parada. Após os canais serem decididos pelo par de nós, a transmissão dos dados começa e múltiplos canais encontrados durante o sensoriamento devem ser usados. Finalmente, os pacotes de

T-RTS e T-CTS são trocados na sinalização pelo CCC, estipulando o fim da transferência de dados e a liberação do canal para outros usuários primários [39].

4.4.2.2. Protocolos de Acesso Agendado

Os protocolos de acesso agendado necessitam de sincronização da rede, em que o tempo é dividido em períodos para canal de controle e transmissão dos dados. Os autores de [17] analisaram o desempenho de um protocolo MAC para redes infraestruturadas, com acesso agendado, utilizando o compartilhamento espectral em faixas de TV e as características do padrão IEEE 802.22 [16, 37]. A estação radiobase gerencia sua própria célula e os usuários secundários. No enlace de descida (DS – *downstream*) a multiplexação por divisão no tempo é utilizada, e no enlace de subida (US – *upstream*) um esquema TDMA por demanda é utilizado. O padrão especifica uma operação de agendamento temporal, com uma hierarquia de quadro de acordo com o exposto na Figura 4.11. Uma hierarquia de *superframes* é definida, em que cada um é formado por vários *frames* MAC, precedidos pelo preâmbulo. No início de cada *superframe* existe um cabeçalho de controle que é usado para informar os usuários secundários sobre os canais disponíveis correspondentes, as larguras de banda definidas, períodos futuros de acesso espectral, entre outros.

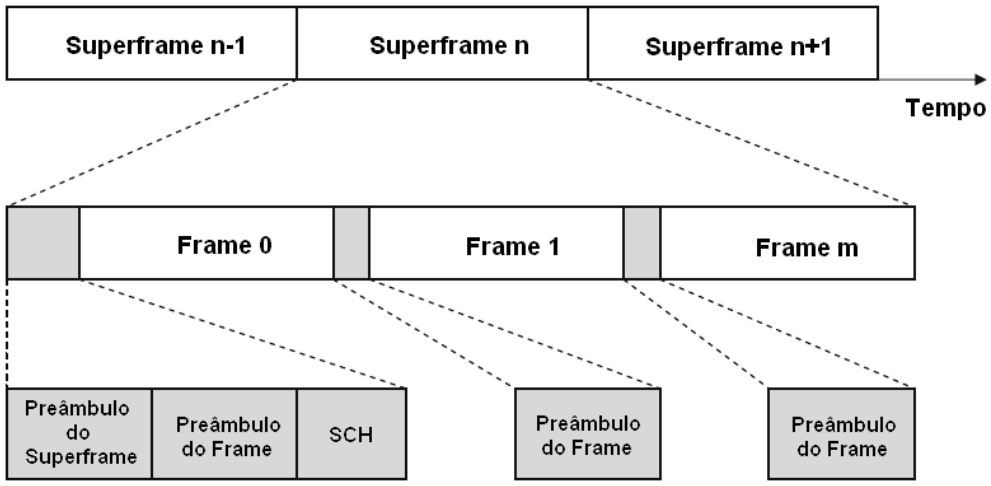


Figura 4.11. Estrutura do *superframe* no IEEE 802.22 [18].

O protocolo *Cognitive* MAC (C-MAC) [15], para redes *ad hoc*, é um esquema sincronizado, agendado temporalmente que apresenta uma considerável vazão de dados (*throughput*) e robustez às mudanças espectrais usando múltiplos transceptores. O C-MAC inclui dois conceitos importantes: o canal Rendezvous (RC – *Rendezvous Channel*) e o canal de apoio (BC – *Backup Channel*). O RC é estabelecido como o canal que pode ser usado para o maior tempo da rede, sem interrupção entre outras escolhas disponíveis. É utilizado para a coordenação dos nós, detecção de usuários primários e alocação de recursos do canal. O BC é determinado pelas medições externas à banda e é usado para prover imediatamente uma escolha de faixas espectrais alternadas no caso do surgimento de um usuário primário.

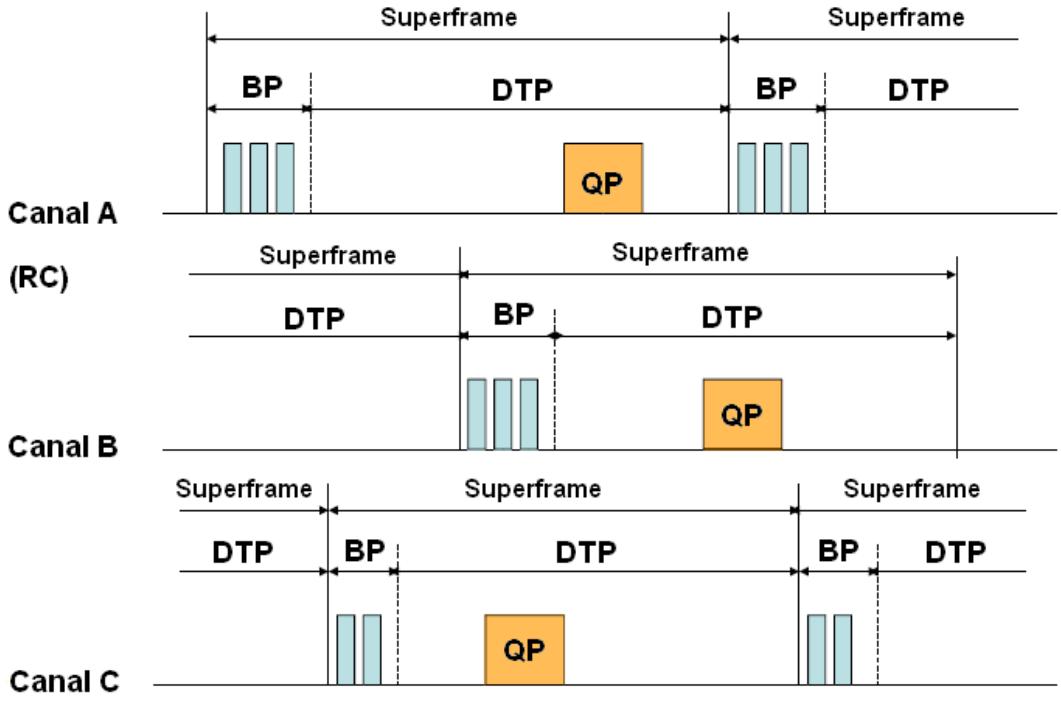


Figura 4.12. A estrutura do *superframe* no C-MAC [18].

No C-MAC, cada faixa espectral possui *superframes* recorrentes, compostos de um período de sinalização (BP – *Beacon Period*) e um período de transferência de dados (DTP – *Data Transfer Period*), conforme mostrado na Figura 4.12. O RC é usado em comunicações de redes de larga abrangência, assim como na descoberta de vizinhos e compartilhamento de informações de carga para cada faixa. Além disso, é utilizado também para a troca dos agendamentos do BP, de modo que as sinalizações não são enviadas simultaneamente sobre todas as faixas espectrais. Nessas faixas, se um usuário cognitivo detecta uma sinalização, então ele pode escolher essa faixa específica e também estabelecer o RC global para a faixa específica de sinalização. A operação do C-MAC é baseada nos seguintes passos [18, 15]:

- Sinalização Distribuída – Cada BP é agendado temporalmente, de modo que usuários cognitvos individuais podem propagar suas sinalizações sem interferência. Pela redifusão das informações das sinalizações recebidas em seus próprios espaços de sinalização, um usuário secundário ajuda a informar seus vizinhos dos outros dispositivos que estão presentes a uma distância maior que o alcance de transmissão;

- Coordenação entre Canais – Os usuários secundários periodicamente se adaptam ao RC e sinalizam sua presença. Se eles precisam estabelecer uma nova faixa de espectro, isso é sinalizado. Qualquer mudança espectral que ocorra no C-MAC deve ser primeiro anunciada pelos usuários cognitivos sobre o RC, antes de utilizar a faixa. Adicionalmente, a adaptação periódica para o RC permite que usuários cognitivos atualizem o sincronismo e obtenham as informações recentes de topologia de vizinhança. A estrutura de *superframe* é então usada na nova faixa espectral;

- Coexistência – A natureza de agendamento temporal do protocolo permite o estabelecimento de períodos de silêncio (QP – *Quiet Periods*) para cada uma das faixas espectrais. Isso garante que os usuários primários sejam diferenciados dos usuários cognitivos e, assim, corretamente detectados. Além disso, as sinalizações são transmitidas com modulação e codificação robustas, de modo que os pacotes que sinalizam a presença de usuários primários são recebidos de forma confiável. Nesse período, uma das faixas espectrais provenientes do BC é escolhida;

- Equilíbrio de Carga – O mecanismo de equilíbrio de carga do C-MAC é alcançado pela acumulação das estatísticas de carga oriundas da análise da sinalização, que transporta a informação de reserva de tráfego dos nós para o *superframe* correspondente.

4.4.2.3. Protocolos de Acesso Híbrido

Os protocolos de acesso híbrido utilizam transmissões parcialmente agendadas, em que a sinalização de controle ocorre geralmente em períodos sincronizados. Entretanto, a transmissão de dados possui esquemas de acesso aleatório ao canal, sem sincronização temporal. Em uma abordagem diferenciada, as fases de controle para transferência de dados possuem durações pré-definidas constituindo um *superframe* comum a todos os usuários da rede. Dentro de cada fase de controle e transferência dos dados o acesso ao canal pode ser aleatório [18].

4.4.3. Desafios Relativos ao Controle de Acesso ao Meio

Existem vários desafios para o acesso eficiente ao espectro. Entre eles, podem ser citados:

- Projeto do Canal de Controle – O acesso ao espectro envolve a sinalização de controle entre os dois usuários secundários em ambas extremidades do enlace. Essa sinalização deve ser ininterrupta pela atividade da vizinhança de usuários primários, uma vez que é usada para a troca de informações sensoriadas e coordenação do acesso ao canal. Por isso, canais de controle com troca dinâmica e confiável precisam ser projetados;

- Adaptação à Transmissão do Usuário Primário – Alguns usuários primários possuem padrões de transmissão específicos, tais como estações de difusão de televisão, ou podem apresentar acessos aleatórios ocasionais ao canal, tais como agências de serviços públicos. Nesses intervalos de tempo, o protocolo MAC cognitivo deve inferir a natureza dos usuários primários e adaptar as transmissões para evitar interferências próprias e prevenir conflitos com usuários primários. Por essa razão, o controle de potência dinâmico e esquemas de agendamento de transmissão também precisam ser projetados.

4.5. Projeto da Camada de Rede

Enquanto os rádios cognitivos estabelecem com sucesso os enlaces para as transmissões sem fio oportunistas, a principal função das redes cognitivas se encontra no projeto da camada de rede, especialmente no roteamento. Isso se deve, primordialmente, ao fato de diversas outras questões de projeto, como controle de fluxo, gerenciamento de recursos de rádio e gerenciamento da mobilidade da rede, serem baseadas nessa funcionalidade. Sendo assim, algumas características relativas ao processo de roteamento em redes cognitivas são apresentadas nesta seção.

Diversos projetos relacionados à inserção de processos cognitivos em esquemas de roteamento têm sido propostos, principalmente a utilização de algoritmos de aprendizagem para técnicas de roteamento em redes cognitivas [75].

Uma aplicação de inteligência artificial em técnicas de roteamento, utilizando um conjunto de agentes inspirados pelo comportamento de colônias de formigas, foi proposta em [40]. Os agentes, que podem ser implementados na forma de pacotes de verificação, exploram a rede com o intuito de coletar informações no atraso fim a fim médio e propagá-las reversamente para atualizar os roteadores intermediários de acordo com as informações coletadas. Os autores de [40] usam trabalhos anteriores relacionados com esquemas de roteamento inspirado em colônias artificiais e aplicam um algoritmo de aprendizagem por reforço, baseado em redes neurais artificiais. As soluções propostas são:

- Uma rede neural artificial é implementada em cada roteador. A rede neural recebe como entrada a probabilidade de selecionar cada possível salto de transmissão seguinte para um determinado nó destino e o tempo médio de viagem para esse nó destino usando cada possível salto de transmissão. As saídas da rede neural são os novos valores de probabilidade e de períodos de viagem estimados para o mesmo nó destino, para os próximos saltos de transmissão possíveis;
- Para cada salto de comunicação, uma formiga de encaminhamento viajando para um determinado nó destino seleciona o próximo salto de comunicação utilizando a rede neural artificial;
- Quando uma formiga viaja em um sentido inverso, a partir do nó destino para um nó visitado previamente, ela atualiza os pesos da rede neural e a tabela de roteamento de acordo com o tempo de viagem medido para o nó destino, modificando o comportamento da rede neural e as escolhas das formigas seguintes.

Os resultados de simulações relatados em [40] mostram que a utilização de técnicas de aprendizagem pode melhorar o desempenho do roteamento, assim como da vazão de dados e uma considerável redução do atraso fim a fim.

Em [29], o esquema de roteamento cognitivo foi apresentado, em que a capacidade de aprendizagem proveniente de um nó da rede é transferida ao pacote de transmissão (pacote cognitivo). Esse pacote é dividido em quatro partes:

- ID, para identificar o pacote e sua classe de serviço;

- DATA, que contém os dados do usuário;
- O campo do mapa cognitivo (CM – *Cognitive Map*);
- O campo do código executável.

Os dois últimos campos são relacionados ao algoritmo de roteamento cognitivo. O CM contém um mapa da rede, isto é, uma estimação do estado da rede baseado em informações prévias coletadas pelo pacote. O código executavel usa o campo CM como uma entrada e um algoritmo de aprendizagem para atualizar o CM. Além disso, o algoritmo de aprendizagem considera um objetivo global pré-definido para o pacote, de modo que uma métrica de desempenho seja otimizada, tal como o atraso mínimo ou a vazão máxima [75].

Os nós da rede apresentam capacidade de armazenamento na forma de caixas de correspondências, que podem ser lidas ou escritas por pacotes cognitivos. Adicionalmente, esses nós processam os códigos executáveis contidos em cada pacote recebido.

Sempre que um pacote cognitivo for recebido por um nó, este executa o código armazenado no campo de código executável do pacote. A entrada do código consiste do mapa cognitivo armazenado no próprio nó e do conteúdo da caixa de correspondência do nó. Como um resultado da execução do código, qualquer uma das seguintes ações é possível:

- O mapa cognitivo do pacote é atualizado;
- A caixa de correspondência do nó é escrita;
- O pacote é enviado por um enlace de saída;
- O pacote é mantido em um armazenador (*buffer*) aguardando que uma determinada condição seja observada.

Os autores comparam o desempenho da rede de pacotes cognitivos com um algoritmo de menor caminho e mostram que mesmo com o uso de algoritmos simples de aprendizagem, o esquema proposto melhora o desempenho da rede, em termos de taxa de perda e atraso. Quando estratégias de aprendizagem mais complexas são implementadas no campo de código executável, tais como redes neurais, a rede apresenta desempenhos ainda melhores [75].

Entretanto, a abordagem proposta em [29] apresenta vários desafios para sua implementação, principalmente em termos do cabeçalho de roteamento devido ao código a ser armazenado em cada pacote. Além disso, os pacotes cognitivos constituem apenas uma pequena parte do total de pacotes e não contêm informações de dados de usuários. Melhoramentos desse trabalho consideraram que a execução dos códigos de aprendizagem são ativados pela chegada dos pacotes cognitivos nos nós [28]. Uma outra versão modificada da solução proposta em [29] aumenta a escalabilidade e reduz o cabeçalho, melhorando ainda mais o desempenho da rede [48].

Uma métrica de roteamento que modela o atraso fim a fim foi proposta pelos autores de [13, 12]. Essa métrica considera tanto o atraso introduzido pelas colisões em uma faixa única de frequência, quanto o atraso introduzido por cada mudança de canal requerida ao longo do percurso [75].

O artigo apresentado em [44] propõe um esquema de espalhamento de informações nas posições dos nós e canais disponíveis para cada nó. Entretanto, o protocolo de troca de informação proposto é testado apenas em um cenário favorável, caracterizado por um canal sem erros e um acesso ao meio sem colisões [75].

O protocolo de roteamento proposto em [76] apresenta uma métrica de roteamento que modela as diferentes características de cada enlace de rádio disponível entre os nós da rede. A métrica é usada para formar uma árvore de roteamento entre a estação radiobase e os nós sem fio na rede. O protocolo considera que a rede é formada por dispositivos utilizando diferentes padrões de comunicação sem fio e que um nó da rede suporta mais de uma interface de rede sem fio [75].

Inspirados no funcionamento do protocolo de roteamento *Ad-hoc On-demand Distance Vector* (AODV) [59], os autores de [10] propuseram o esquema de roteamento sob demanda para redes de rádio cognitivo (CRNO – *Cognitive Radio Networks On-demand*). Cada usuário cognitivo executa o roteamento apenas quando existe uma demanda. A arquitetura da rede apresentada na Figura 4.13 foi considerada para a descrição desse protocolo.

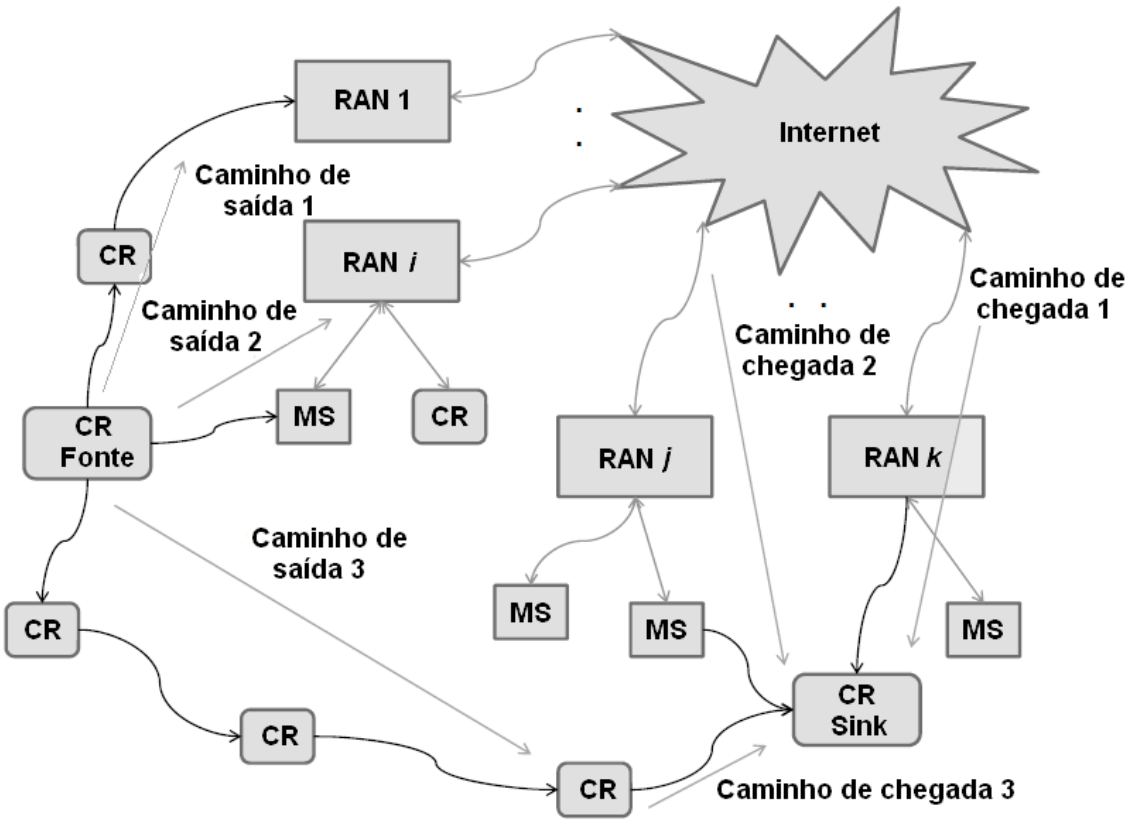


Figura 4.13. O roteamento de pacotes em redes cognitivas [9].

A arquitetura contém uma rede infraestruturada, que pode ser a Internet e várias redes de acesso por rádio (RAN – *Radio Access Networks*) que provêm diversas vias para acesso à rede infraestruturada. Estações móveis (MS – *Mobile Stations*) estão associadadas à determinadas tecnologias de RAN. Cada rádio cognitivo (CR – *Cognitive Radio*) é capaz de configurar seus parâmetros ao sistema de rádio apropriado, com o intuito de transportar pacotes de dados e controle de tráfego. RANs, MSs e a rede infraestruturada podem ser qualquer sistema primário específico. Um rádio cognitivo também pode ser uma estação móvel de um sistema primário. Todos os enlaces nos sistemas primários são bidirecionais e representados por setas de sentido duplo. Enlaces oportunísticos, devido ao acesso espectral dinâmico dos rádios cognitivos, e determinados enlaces *ad hoc* são unidirecionais e representados por setas de sentido único. Existem três caminhos cooperativos diferentes para transportar os pacotes. Conforme pode ser observado na Figura 4.13, os Caminhos de Saída 3 e Caminhos de Chegada 3 geralmente representam redes *relay* de rádios cognitivos (CRRN – *Cognitive Radio Relay Networks*) [10].

O transmissor CR e receptor CR formam um enlace cognitivo, tipicamente usando o acesso espectral dinâmico. O receptor CR pode ser um rádio cognitivo ou um nó no sistema primário. O nó CR Fonte e o nó CR Destino formam um enlace virtual. O nó CR Destino pode ser um rádio cognitivo ou qualquer nó no sistema primário. Se o nó CR Destino for um rádio cognitivo, ele é denominado nó CR Sink.

A mensagem de roteamento inclui um cabeçalho formado pelas seguintes informações [9]:

- O endereço IP do usuário cognitivo de destino;
- O endereço IP do usuário cognitivo fonte;
- A identificação ID da mensagem (msg_id);
- O endereço IP do usuário cognitivo *relay* (cr_relay_ip);
- O endereço IP do transmissor cognitivo (cr_tx_ip) e seu tipo de rádio (cr_tx_type) para o pacote recebido;
- O endereço IP do receptor cognitivo (cr_rx_ip) e seu tipo de rádio (cr_rx_type) para o pacote encaminhado;
- O número da sequência (seq_count) associado com o percurso (cr_tx_ip, cr_relay_ip, cr_rx_ip), iniciando de 0 (zero) e adicionando 1 (um) para cada caminho igual;
- O contador de tempo para cada usuário cognitivo *relay* (time_counter), iniciando de 0 (zero) e adicionando 1 (um) para uma novo período de tempo.

Quando um usuário cognitivo ou um terminal móvel do sistema primário surge no ambiente de rádio, a aquisição de seu endereço IP pode não ser viável de imediato e uma ID pode ser utilizada para servir a proposta da tabela. O roteamento CRNO é ilustrado na Figura 4.14 e consiste de três fases de operação [10]:

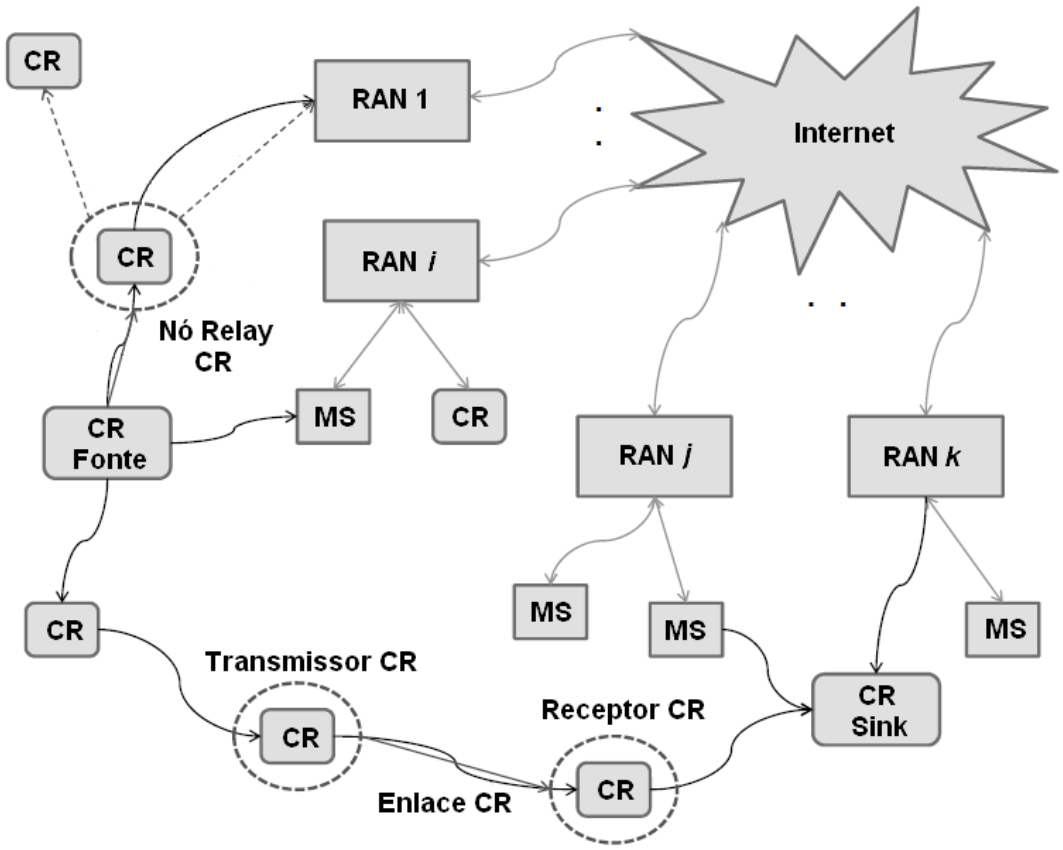


Figura 4.14. O roteamento em redes CRNO [9].

- Fase de Sensoriamento – O usuário cognitivo sensoria o espectro de múltiplos sistemas coexistentes (e possivelmente diferentes faixas de frequência), para atualizar sua tabela de roteamento. Essa tabela armazena informações reconhecendo cada receptor cognitivo em potencial, histórico, estimação de confiança no usuário cognitivo e parâmetros de comunicação para ajustar as configurações do equipamento de rádio. Cada receptor cognitivo em potencial é identificado por um endereço IP que pode ser adquirido a partir de suas transmissões passadas ou por uma ID designada pelo usuário cognitivo. O histórico pode ser uma sinalização simples para indicar se o receptor cognitivo é confiável ou não, baseado em processos de aprendizagem. Os parâmetros de comunicação são obtidos a partir do sensoriamento espectral para o ajuste do rádio cognitivo.

- Fase de Descoberta de Rotas – No momento em que o usuário cognitivo origina um pacote para o destino ou recebe um pacote de um nó *relay*, ele busca alguma violação na tabela de roteamento. Se violações não forem constatadas, o nó cognitivo seleciona outro nó cognitivo a partir da tabela de encaminhamento de rotas. Os enlaces para o sistema primário possuem maior prioridade. Por outro lado, quando uma violação ocorre, o nó *relay* cognitivo busca uma oportunidade para reconhecer negativamente o transmissor cognitivo, baseado na tabela de roteamento. O nó transmissor cognitivo tenta re-rotear o pacote para outro nó *relay* cognitivo (se possível), ou ainda transmitir o pacote de volta caso não existam rotas disponíveis.

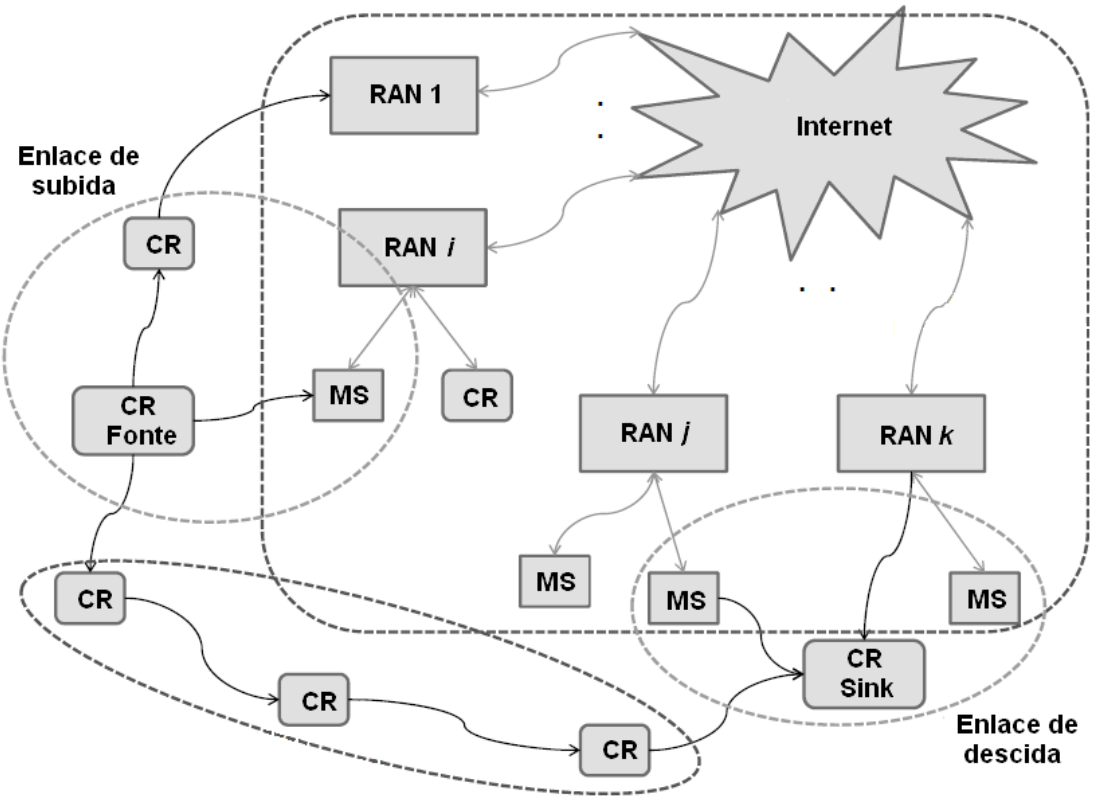


Figura 4.15. Segmentação ou decomposição de uma rede cognitiva [9].

- Fase de Atualização da Tabela – Além da seleção do enlace para completar o roteamento, uma rota reversa associada a esse *relay* precisa ser atualizada como uma parte da tabela de rotas reversas. Cada rota reversa consiste de parâmetros msg_id, cr_rx_ip, cr_rx_type, cr_tx_ip, cr_tx_type e seq_count. Ambos cr_rx_type e cr_tx_type especificam a operação de coexistência de sistemas multi-rádio nas redes cognitivas.

Para que os protocolos de roteamento cognitivo por demanda funcionem satisfatoriamente é necessário implementar o fluxo de controle na camada de rede. Em contraste com os esquemas convencionais de fluxo de controle em redes de computadores, o objetivo principal para redes cognivas é atender o controle de danos [10].

Além disso, é possível observar que a arquitetura geral de uma rede cognitiva pode ser segmentada em diversas partes, conforme mostrado na Figura 4.15. Nesse esquema, os pacotes são roteados a partir do nó cognitivo fonte para o nó cognitivo destino, por meio dos seguintes segmentos [9]:

- Enlace de subida da rede cognitiva;
- Sistemas primários multi-rádio coexistentes, normalmente com uma arquitetura infraestruturada (tal como a Internet), o que pode ser considerado como um tipo de tunelamento em redes cognitivas para transportar os pacotes rapidamente;

- O enlace de descida da rede cognitiva;
- A rede *relay* cognitiva.

A rede *relay* cognitiva é considerada como um tipo especial de rede cognitiva, em que os usuários cognitivos cooperam (*relaying*) na transmissão dos pacotes.

O fluxo de dados é caracterizado de duas maneiras [10]:

- Nó cognitivo fonte$\longrightarrow$ Enlace de subida da rede cognitiva $\longrightarrow$ Sistema primário e infraestrutura $\longrightarrow$ Enlace de descida da rede cognitiva $\longrightarrow$ Nó cognitivo destino;
- Nó cognitivo fonte$\longrightarrow$ CRRN $\longrightarrow$ Nó cognitivo destino.

Para o enlace de subida da rede cognitiva, o roteamento tenta atingir o sistema primário por meio de enlaces oportunísticos. Por exemplo: na Figura 4.15, quando o nó *relay* cognitivo está no processo de seleção da rota de encaminhamento, ele tende a selecionar o nó mais próximo ao sistema primário, que é o nó na RAN 1. O roteamento tenta deixar o sistema primário por meio de enlaces cognitivos oportunísticos para o enlace de descida da rede cognitiva.

4.5.1. Desafios Relativos ao Projeto da Camada de Rede

Apesar de existirem algumas propostas de esquemas de roteamento, diversos desafios relacionados com o projeto de camada de rede são discutidos a seguir [9]:

- Disponibilidade de Enlace – os enlaces de redes cognitivas ficam disponíveis em períodos determinados pelo sistema primário, de modo que o acesso dinâmico ao espectro pode efetivamente aproveitar tais oportunidades, após um sensoriamento espectral bem sucedido. Desse modo, enlaces de redes cognitivas, especialmente os que envolvem usuários cognitivos transmissores e receptores, ficam disponíveis aleatoriamente, o que resulta em uma topologia da rede cognitiva aleatória. As condições dos enlaces de redes cognitivas variam muito, pois a duração da disponibilidade do enlace é apenas da ordem de milisegundos, em vez de segundos, minutos, horas, ou mesmo dias, conforme em outros tipos de redes sem fio;
- Enlaces Unidirecionais – Enlaces unidirecionais não são comuns em redes sem fio típicas, que geralmente possuem enlaces bidirecionais (a comunicação de rádio normalmente é *half-duplex*). Em redes de sensores sem fio e redes *ad hoc*, enlaces unidirecionais são possíveis devido à potência de transmissão assimétrica e diferentes níveis de interferência nos receptores. Entretanto, os enlaces unidirecionais são mais prováveis em redes cognitivas, devido ao fato de que um terminal com rádio cognitivo pode apenas ter oportunidades de transmissão em intervalos de tempo restrito e não há garantias de permissão para transmitir a partir de outra via de comunicação. Além disso, um usuário cognitivo pode influenciar um usuário primário para (cooperativamente) transmitir os pacotes, mas a outra via de comunicação pode não ser permitida e vice-versa;

- Redes Sem Fio Heterogêneas – as redes cognitivas são formadas por redes sem fio heterogêneas. O *handoff* dos nós cognitivos é normalmente necessário para o roteamento nessas redes. Entretanto, os enlaces de redes cognitivas devem estar disponíveis por uma curta duração de tempo e a cooperação entre os nós da rede interfere significativamente no desempenho satisfatório da rede;

- Re-roteamento – Em redes cognitivas, devido à conectividade intermitente, uma rota estabelecida para o fluxo de dados pode mudar devido à disponibilidade espectral e à mobilidade da rede. Portanto, algoritmos de re-roteamento que considerem a característica dinâmica do espectro são necessários. Um esquema de roteamento ciente do espectro deve adaptar as seleções das rotas às flutuações do espectro [2];

- Gerenciamento de Filas – Um usuário cognitivo deve possuir múltiplas interfaces para a comunicação com diferentes nós. Dado que a disponibilidade do espectro varia com o tempo, essas interfaces podem se tornar indisponíveis, requerendo que os pacotes servidos por uma interface sejam transferidos para outra. Além disso, os requisitos de qualidade de serviço podem apresentar várias prioridades em tipos de tráfego diferentes. Portanto, a implementação de um modelo de fila única ou de múltiplas filas para cada tipo de tráfego de cada interface precisa ser investigada [2].

4.6. Aplicações de Redes Cognitivas

As redes cognitivas são um novo paradigma em comunicações sem fio, que tendem a disponibilizar melhores serviços para diversos nichos de mercado. Os rádios cognitivos, que integram essas redes, podem perceber o ambiente de propagação, aprender padrões e adaptarem seus parâmetros para atender a requisitos imediatos do usuário, da rede e do ambiente de rádio [4]. Entre as várias áreas de aplicações das redes cognitivas, podem ser citadas [2]:

- Redes Alugadas – A rede primária pode prover uma rede alugada permitindo o acesso oportunista do espectro licenciado, por meio do acordo com a rede secundária em não prejudicar os parâmetros de qualidade de serviço dos usuários primários [69]. Por exemplo, a rede primária pode alugar o acesso ao espectro apenas para uma operadora. A rede primária também pode disponibilizar o acesso ao espectro para uma comunidade regional, com o propósito de prover acesso sem fio por banda larga;

- Redes *Mesh* Cognitivas – Redes *mesh* sem fio têm surgido como uma tecnologia de baixo custo para prover conectividade em banda larga [3]. Entretanto, à medida que a densidade da rede aumenta e as aplicações demandam uma maior vazão de dados, as redes *mesh* necessitam de um aumento em sua capacidade para atender os requisitos das aplicações. Considerando que a tecnologia de rádio cognitivo proporciona o acesso a faixas mais largas do espectro, as redes cognitivas podem ser utilizadas para redes *mesh* que serão implementadas em áreas urbanas densas [47]. Por exemplo, a área de cobertura de redes cognitivas pode aumentar se um *backbone mesh* sem fio é estabelecido baseado em pontos de acesso cognitivos e nós *relay* cognitivos [5]. A capacidade de um ponto de acesso cognitivo, conectado pelo acesso em

banda larga à Internet, é distribuida em uma extensa área com o auxílio de nós *relay* cognitivos. As redes cognitivas têm a capacidade de acrescentar, de maneira temporaária ou permanente, alocações espectrais para os enlaces usados nos esquemas de transmissão cooperativa (*relaying*) no caso de alta carga de tráfego;

- Redes de Emergência – As redes cognitivas também podem ser implementadas para o auxílio na operação de redes de emergência [51]. No caso de desastres naturais, que podem temporariamente inviabilizar a infraestrutura de comunicação existente, as ações das equipes de emergência nas áreas do desastre precisam formar redes de emergência. Considerando que as redes de emergência lidam com informações críticas, uma comunicação segura precisa ser garantida com a mínima latência de transmissão. Além disso, comunicações em situações de emergência requerem uma disponibilidade significante de espectro de rádio para a manipulação de grandes volumes de tráfego de dados, incluindo voz, vídeo e dados. As redes cognitivas podem oferecer o uso eficaz do espectro existente sem a necessidade de uma infraestrutura e mantendo as prioridades de comunicações e tempos de resposta;

- Redes Militares – Uma das aplicações das redes cognitivas está no ambiente de comunicações militares. As redes cognitivas permitem aos dispositivos militares de comunicação escolher faixas de frequência intermediária, esquemas de modulação e de codificação adaptáveis às variações do ambiente de rádio em campos de batalha. Adicionalmente, as redes militares demandam recursos de segurança e proteção das transmissões em ambientes hostis. As redes cognitivas permitem às equipes militares realizarem o *handoff* espectral e identificar faixas espectrais seguras, livres de interceptação por tropas inimigas [2];

- Segurança Pública – A área de segurança pública é outra área em que as redes cognitivas têm mostrado potencial para aplicação. Durante anos, as agências de segurança pública têm necessitado de alocação espectral adicional para solucionar o congestionamento de faixas de frequência. Por meio dos benefícios das técnicas de compartilhamento espectral, as redes cognitivas podem utilizar algumas das faixas espectrais existentes e pouco utilizadas, enquanto mantêm a prioridade de solicitações de atendimento e tempo de resposta. Além disso, as redes cognitivas podem melhorar a interoperabilidade provendo enlaces de comunicação entre diferentes jurisdições [51];

- Serviços – O ramo de serviços possui diversas oportunidades para o uso de redes cognitivas. Um exemplo dessa aplicação é utilizar redes cognitivas para melhorar os serviços de comunicação de um hotel em que ocorre uma conferência [4]. Suponha que o hotel utilize uma rede no padrão IEEE 802.11, em que o custo para utilização da rede sem fio está inclusa no preço dos serviços da conferência e serviços de quarto. A rede de comunicação experimenta demandas de acesso dos palestrantes e outros inscritos na conferência, assim como de outras outras pessoas que estejam hospedadas com outras finalidades. Sem restrições de utilização, qualquer pessoa com um dispositivo compatível com o padrão IEEE 802.11 possui o mesmo potencial para acessar a Internet. Em períodos de alta demanda, todos os usuários podem experimentar serviços lentos e interrupções, o que leva à insatisfação dos usuários.

Uma solução seria restringir o acesso à rede e cobrar uma taxa. As taxas poderiam variar de acordo com a categoria do usuário:

- Taxas pagas pela organização da conferência, cujo o custo é repassado aos inscritos no pacote de inscrição;
- Hóspedes instalados em quartos com desconto deveriam pagar taxas maiores;
- Hóspedes instalados em quartos *premium* teriam direito a acessar a rede livremente, sem pagar taxas por isso.

Entretanto, sem um sistema de priorização de usuários, os problemas de qualidade de serviço persistirão. Os participantes da conferência ficarão frustrados caso um palestrante tente conduzir uma demonstração em tempo real, usando a Internet, e experimente uma degradação na qualidade de seu acesso à rede. Para proporcionar a satisfação com a qualidade dos serviços oferecidos, o hotel precisa disponibilizar uma alternativa em que o palestrante tenha acesso aos recursos que são necessários para conduzir uma apresentação suave e contínua. Uma outra solução seria o estabelecimento de duas redes, uma das quais, restrita a um número reduzido de apresentadores e usuários prioritários.

Redes cognitivas representam uma solução eficaz para o problema, mesmo no caso em que o serviço de acesso à Internet é disponibilizado usando uma única frequência de rádio. Cada usuário poderia ser vinculado a um nível de prioridade de uso dos serviços, baseado nas metas de serviço do hotel. Nesse caso, o rádio cognitivo otimizaria o acesso à rede, de modo que usuários com um nível de prioridade mais alta tivessem preferência no acesso em relação aos usuários com um nível de prioridade mais baixa. Uma prioridade mais alta poderia ser atribuída a um número limitado de palestrantes da conferência para garantir que suas apresentações ocorram de forma contínua. Por outro lado, os hóspedes que não estão registrados na conferência possuiriam um acesso razoável no andar de seus quartos, mas apenas às lacunas do espectro nas áreas da conferência.

4.7. O Padrão IEEE 802.22

O IEEE 802.22 é o primeiro padrão a contemplar a utilização da tecnologia de rádio cognitivo [56]. Ele foi proposto no contexto do grupo de trabalho 802.22 relativo à redes sem fio regionais (WRAN – *Wireless Regional Area Network*), para o uso não licenciado de faixas de TV para comunicações sem fio. Nos Estados Unidos, três tipos de sinais de TV devem ser detectados: sinais de TV analógica (NTSC – *National Television System Committee*), de TV digital (ATSC – *Advanced Television Systems Committee*) e de microfones sem fio [64].

Uma rede IEEE 802.22 é um tipo de rede sem fio infraestruturada onde uma estação radiobase (BS – *base station*) coordena o acesso de diversos nós em uma célula de um único salto de comunicação (*single hop cell*). Essa célula cobre uma área com o raio que varia entre 33 km (típico) a 100 km [43]. Os usuários em uma célula 802.22 são chamados de *Consumer Premise Equipments* (CPEs).

Os equipamentos WRAN apresentam uma potência de transmissão muito superior à dos demais sistemas IEEE 802 (a potência dos CPEs podem alcançar até 4 W, enquanto

que as estações base atingem o máximo de 100 W) [14]. Além disso, as faixas de TV apresentam boas características de propagação, permitindo que a área de cobertura atinja até 100 km.

A tecnologia 802.22 permite a reutilização das bandas UHF/VHF, em que três tipos de sinais primários podem estar presentes nestas bandas: sinais de TV analógica, digital e microfones sem fio. Considerando o valor mínimo da razão de potência desejada-indesejada de sinal (D/U – *Desired-to-Undesired signal power ratio*[2]) de 23 dB e o contorno de proteção de 134,2 km, o raio de afastamento dos CPEs do transmissor de TV deve ser dado por 150,3 km [65]. Os CPEs dentro desse raio de afastamento são forçados a evitar o uso do canal de TV. A Figura 4.16 ilustra esse cenário.

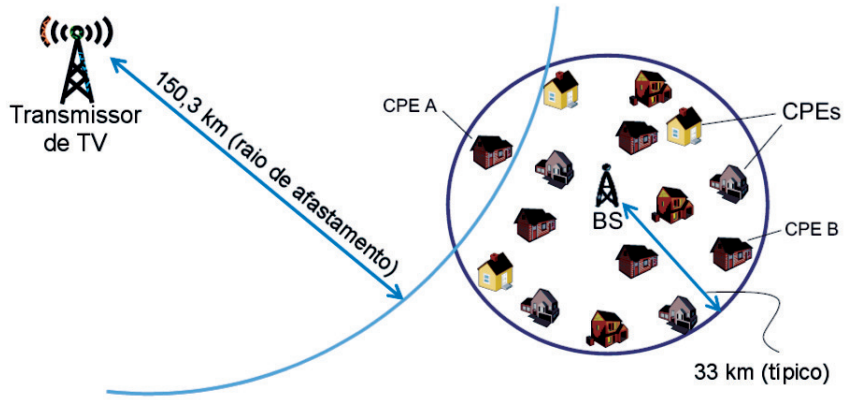


Figura 4.16. Ilustração de uma célula IEEE 802.22 coexistindo com um transmissor de TV [43].

4.7.1. Modelo de Sensoriamento do Canal no Padrão IEEE 802.22

No padrão IEEE 802.22 o canal é modelado como uma fonte LIGADA / DESLIGADA (*ON/OFF*), onde um período *ON* representa o tempo de duração em que os usuários primários estão utilizando ativamente seus respectivos canais. Dessa forma, os usuários secundários somente têm permissão para utilizar o canal nos períodos *OFF* dos usuários primários. Esse modelo tem sido utilizado com sucesso na modelagem do padrão de uso do canal dos usuários primários em várias aplicações [27, 42, 57]. Um padrão de uso de canal por transmissores de TV usualmente apresentam períodos muitos longos de *ON* e *OFF* (da ordem de horas).

Por meio do sensoriamento espectral o rádio cognitivo verifica o estado do canal durante o tempo de sensoriamento (denotado por T_I – *sensing-time*) e detecta a presença de sinais primários no momento. O valor de T_I pode variar de acordo com o método de detecção utilizado (*e.g.,* menos de 1 ms para detecção de energia). A Figura 4.17 ilustra o modelo de canal *ON/OFF* e um exemplo do processo de sensoriamento periódico com o tempo de sensoriamento T_I e o período de sensoriamento (denotado por *sensing-period* – T_P) [43]. Esse último pode ser utilizado para obter a frequência de sensoriamento ($1/T_P$ – *sensing-frequency*).

No padrão 802.22, o sensoriamento pode ser realizado durante os períodos de

[2]A razão entre o sinal desejado e os sinais indesejados (soma de todos os sinais interferentes).

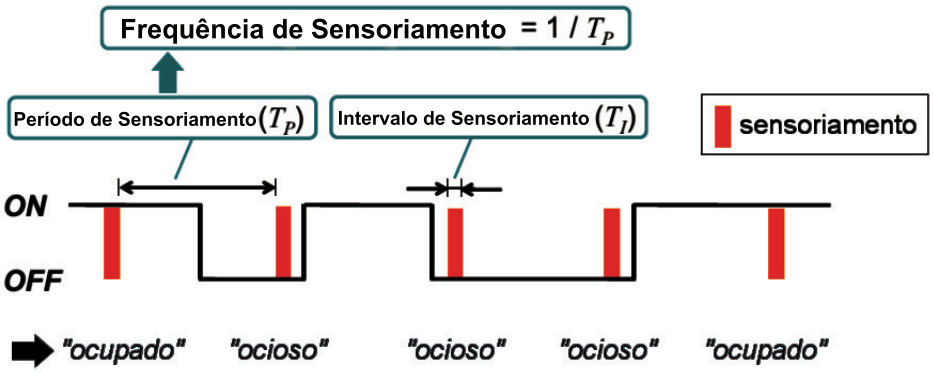


Figura 4.17. O modelo de canal *ON/OFF* e o processo de sensoriamento periódico [43].

silêncio (*i.e.,* enquanto as comunicações entre os usuários secundários estão suspensas), dentro dos quais nenhum CPE tem permissão para transmitir dados. Caso seja realizado um sensoriamento colaborativo, os períodos de silêncio passam a ser sincronizados entre os diversos sensores de uma mesma célula, bem como entre células vizinhas, o que é possível por meio de um protocolo de sinalização de coexistência (CBP – *Coexistence Beacon Protocol*), via a troca de quadros de informação [17].

4.7.2. Requisitos de Sensoriamento Espectral do Padrão IEEE 802.22

A funcionalidade de sensoriamento espectral é uma parte crítica do padrão IEEE 802.22 [63]. Por meio dela é possível detectar a presença ou ausência de um usuário primário licenciado e tomar a ação apropriada de liberar a faixa de espectro em uso ou continuar utilizando-a. Caso necessário, o CPE deve liberar a faixa no período de 2 segundos a partir do início da transmissão do usuário primário. A Tabela 4.3 apresenta os requisitos de sensoriamento espectral do padrão IEEE 802.22 [67].

Tabela 4.3. Requisitos de sensoriamento espectral [46].

Parâmetro	Valor de detecção
Tempo de detecção do canal	$\leq$ 2 seg
Tempo de mudança do canal	2 seg
Tempo de encerramento da transmissão no canal	100 mseg

O limiar de detecção incumbente (IDT – *Incumbent Detection Threshold*) é a menor potência de sinal primário (em dBm) que os CPEs devem detectar [46]. Na Tabela 4.4 são apresentados os valores de IDT para os três tipos de sinais primários suportados nos Estados Unidos [37].

O tempo de detecção do canal (CDT – *Channel Detection Time*) deve ser de ≤ 2 segundos, período dentro do qual os usuários primários devem ser detectados com probabilidade de erro de detecção menor que 0,1, independentemente do número de vezes que o sensoriamento seja realizado durante o CDT. Da mesma forma, o valor da probabilidade de falso alarme deve ser inferior a 0,1 quando o mesmo algoritmo de sensoriamento executa por CDT segundos, período durante o qual nenhum usuário primário está presente. O requisito da probabilidade de erro de detecção serve para garantir a mínima interferência

Tabela 4.4. Limiar de detecção incumbente (IDT) dos sinais primários [43].

Tipo de sinal	IDT
TV analógica (NTSC)	-94 dBm (no pico de sincronização da portadora de imagem do NTSC)
TV digital (ATSC)	-116 dBm (largura de banda de 6 MHz)
Microfones sem fio	-107 dBm (largura de banda de 200 kHz)

com outros usuários, enquanto que o requisito da probabilidade de falso alarme serve para evitar mudanças desnecessárias de canal devido à falsa detecção de usuários primários.

O IEEE 802.22 também provê um mecanismo chamado sensoriamento em dois estágios (TSS – *two-stage sensing*), em que o algoritmo de sensoriamento pode decidir qual técnica de sensoriamento (detecção de energia ou detecção de característica) deve ser utilizada em um período de silêncio. Embora a detecção de energia seja responsável por um *overhead* temporal mínimo (usualmente menor que 1 ms), ela é suscetível às incertezas sobre o ruído [62]. A detecção de características é menos suscetível às incertezas sobre o ruído [35], mas requer um maior tempo de sensoriamento (*e.g.,* 24,2 ms para a detecção de campos de sincronização de sinais ATSC [17]).

4.7.3. O Mecanismo TSS do Padrão IEEE 802.22

Para ajudar o algoritmo de sensoriamento a atingir os requisitos de deteção apresentados na Seção 4.7.2, o IEEE 802.22 provê um mecanismo de sensoriamento em dois estágios (TSS – *two-stage sensing*). Com o TSS, o algoritmo de sensoriamento pode escalonar um sensoriamento rápido ou um sensoriamento fino em cada período de silêncio (QP – *quiet period*). O sensoriamento rápido emprega detecção de energia enquanto o sensoriamento fino utiliza detecção de características [43].

Embora um algoritmo de sensoriamento possa escalonar quantos QPs forem necessários, existem restrições sobre o período de sensoriamento. Por exemplo, o QP de um sensoriamento rápido, usualmente menor que 1 ms, pode ser escalonado no fim de um quadro MAC 802.22 (de duração de 10 ms), no máximo uma vez a cada quadro. Consequentemente, o período de sensoriamento rápido é um múltiplo do tamanho do quadro (*i.e.,* $n \times 10$ ms, em que n é um inteiro positivo). Por outro lado, a duração do QP de um sensoriamento fino varia de acordo com o esquema de detecção de característica utilizado. No caso do esquema de detecção de característica requerer um tempo sensoriamento maior que um quadro MAC, seu QP deve ser escalonado entre quadros MAC consecutivos [43].

4.8. Considerações finais

As redes cognitivas foram desenvolvidas como uma tentativa de resolver os problemas relativos à limitada disponibilidade de faixas de espectro e à ineficiência na utilização destes. Essas redes são equipadas com as capacidades intrínsecas dos rádios cognitivos e oferecem um paradigma de comunicações ciente sobre a ocupação do espectro em redes sem fio. Para tanto, é necessário que o processo de sensoriamento espectral seja eficaz e determine, com a maior confiabilidade possível, estimativas dos parâmetros do

espectro.

Diversas técnicas podem ser empregadas para o sensoriamento espectral. Cada uma dessas técnicas é mais eficaz em determinadas circunstâncias, e requer diferentes níveis de complexidade de implementação. Técnicas de processamento de sinais podem ser aplicadas durante esse processo. Outra maneira de melhorar o sensoriamento é utilizar estratégias cooperativas, em que diversos rádios cognitivos compartilham seus resultados, criando um mapa global de ocupação do espectro.

Os resultados da fase de sensoriamento do canal e detecção de interferência, obtidos a partir da camada física, podem ser usados pela camada MAC para formar um histórico de ocupação do canal no tempo. As técnicas de acesso ao espectro em redes cognitivas, classificadas em protocolos de acesso aleatório, agendado e híbrido, ainda não integram completamente a função de sensoriamento ao controle de acesso ao meio. Além disso, novas métricas de desempenho de protocolos MAC, que capturem características específicas do contexto de redes cognitivas, precisam ser propostas.

Diversas questões de projeto, como controle de fluxo, gerenciamento de recursos de rádio e mobilidade da rede são baseadas no protocolo de roteamento adotado. Alguns esquemas de roteamento envolvendo algoritmos de aprendizagem foram discutidos neste capítulo, assim como técnicas de roteamento sob demanda para redes cognitivas. A grande variação de condições dos enlaces cognitivos, necessidades de esquemas de reroteamento e de desenvolvimento de modelos de alto desempenho para o gerenciamento de filas em redes cognitivas constituem desafios significativos para o projeto da camada de rede.

Algumas características do padrão IEEE 802.22 também foram discutidas neste capítulo. Ele utiliza a tecnologia de redes cognitivas para transmissões em redes sem fio regionais, por meio do acesso não licenciado ao espectro de TV. O IEEE 802.22 é o primeiro padrão mundial baseado na tecnologia de rádio cognitivo [16, 37]. Canais de TV específicos, assim como bandas de guarda serão utilizados para a comunicação nesse padrão.

No entanto, as agências reguladoras deverão estabelecer diversos parâmetros para o funcionamento das redes cognitivas, dado que os usuários destas podem vir a causar interferências nas transmissões dos sistemas primários. Para tanto, diversos desafios relacionados à incertezas nas comunicações geram impactos diretos sobre o desempenho das técnicas de sensoriamento, podendo levar até mesmo a interferências com usuários primários.

Este capítulo apresentou diversos aspectos e desafios relativos ao projeto e implementação de redes cognitivas. Espera-se que ele sirva como base para os primeiros estudos sobre o assunto, agregando conceitos e resultados desenvolvidos em diversos trabalhos científicos recentes e de bastante relevância.

Agradecimentos

Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e à Fundação de Amparo à Pesquisa e ao Desenvolvimento Científico e Tecnológico do Maranhão (FAPEMA) pelas bolsas de doutorado de Marcelo

Portela Sousa e Rafael Fernandes Lopes, respectivamente. Adicionalmente, os autores agradecem à infraestrutura de pesquisa provida pelo Instituto de Estudos Avançados em Comunicações (IECOM) e pela Universidade Federal de Campina Grande (UFCG).

Referências

[1] I. Akyildiz, Y. Altunbasak, F. Fekri, and R. Sivakumar. Adaptnet: An adaptive protocol suite for the next-generation wireless internet. *IEEE Communication Magazine*, 3(42):pp.128–138, 2004.

[2] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty. Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*, (50):pp.2127–2159, 2006.

[3] I. Akyildiz and X. Wang. A survey on wireless mesh networks. *Communications Magazine, IEEE*, 43(9):S23 – S30, sept. 2005.

[4] S. Ball and A. Ferguson. Consumer applications of cognitive radio defined networks. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages 518 –525, nov. 2005.

[5] L. Berlemann, S. Mangold, and B. Walke. Policy-based reasoning for spectrum sharing in radio networks. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages 1 –10, nov. 2005.

[6] T. X. Brown. An analysis of unlicensed device operation in licensed broadcast service bands. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.11–29, Baltimore, Nov. 2005.

[7] M. Buddhikot, P. Kolodzy, S. Miller, K. Ryan, and J. Evans. Dimsumnet: new directions in wireless networking using coordinated dynamic spectrum. In *Sixth IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005*, pages pp.78–85, June 2005.

[8] D. Cabric, S. M. Mishra, and R. W. Brodersen. Implementation issues in spectrum sensing for cognitive radios. In *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*, 2004.

[9] K. Chen and R. Prasad. *Cognitive Radio Networks*. John Wiley and Sons, 2009.

[10] K.-C. Chen, B. K. Cetin, Y.-C. Peng, N. Prasad, J. Wang, and S. Lee. Routing for cognitive radio networks consisting of opportunistic links. In *Wireless Communications and Mobile Computing*, volume 10, pages 451–466, march 2009.

[11] K.-C. Chen, Y.-J. Peng, N. Prasad, Y.-C. Liang, and S. Sun. Cognitive radio network architecture: Part I – general structure. In *ICUIMC '08: Proceedings of the 2nd international conference on Ubiquitous information management and communication*, pages pp.114–119, New York, NY, USA, 2008. ACM.

[12] G. Cheng, W. Liu, Y. Li, and W. Cheng. Joint on-demand routing and spectrum assignment in cognitive radio networks. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 6499 –6503, june 2007.

[13] G. Cheng, W. Liu, Y. Li, and W. Cheng. Spectrum aware on-demand routing in cognitive radio networks. In *New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007. 2nd IEEE International Symposium on*, pages 571 –574, april 2007.

[14] G. Chouinard. WRAN reference model, Jan. 2007. doc.: IEEE 802.22-04/0002r15.

[15] C. Cordeiro and K. Challapali. C-MAC: A cognitive MAC protocol for multi-channel wireless networks. In *New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007. 2nd IEEE International Symposium on*, pages 147 –157, april 2007.

[16] C. Cordeiro, K. Challapali, D. Birru, and S. Shankar. IEEE 802.22: the first worldwide wireless standard based on cognitive radios. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.328–337, Baltimore, Nov. 2005.

[17] C. Cordeiro, K. Challapali, and M. Ghosh. Cognitive PHY and MAC layers for dynamic spectrum access and sharing of TV bands. In *TAPAS '06: Proceedings of the first international workshop on Technology and policy for accessing spectrum*, page 3, New York, NY, USA, 2006. ACM.

[18] C. Cormio and K. R. Chowdhury. A survey on MAC protocols for cognitive radio networks. *Ad Hoc Networks*, 7(7):pp.1315–1329, 2009.

[19] F. Digham, M. Alouini, and M. Simon. On the energy detection of unknown signals over fading channels. In *Proc. IEEE ICC 2005*, volume vol.5, pages pp.3575–3579, May 2003.

[20] FCC. Et docket no.02-155. Technical report, Spectrum Policy Task Force Report, Nov 2002.

[21] FCC. FCC radio spectrum home page, 2010. Disponível em: `http://www.fcc.gov/oet/spectrum`. Acesso em março de 2010.

[22] A. Fehske, J. D. Gaeddert, and J. H. Reed. A new approach to signal classification using spectral correlation and neural networks. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.144–150, Baltimore, Nov. 2005.

[23] B. Fette. *Cognitive radio technology*. Elsevier Science & Technology Books, 2006.

[24] B. Fette. *Cognitive radio, software defined radio, and adaptive wireless systems*, chapter Introducing Adaptive, Aware, and Cognitive Radios. Springer, 2007.

[25] L. C. Freitas, A. Klautau, and J. C. W. A. Costa. Classificadores de modulação digital em sensoriamento espectral de rádio cognitivo. In *Simpósio Brasileiro de Telecomunicações (SBrT) 2008*, 2008.

[26] W. A. Gardner. Signal interception: A unifying theoretical framework for feature detection. *IEEE Trans. on Communications*, vol.36(8):pp.897–906, August 1988.

[27] S. Geirhofer, L. Tong, and B. M. Sadler. Dynamic spectrum access in the time domain: Modeling and exploiting white space. *IEEE Communications Magazine*, 45(5):pp.66–72, May 2007.

[28] E. Gelenbe and P. Liu. QoS and routing in the cognitive packet network. In *World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a*, pages 517–521, june 2005.

[29] E. Gelenbe, Z. Xu, and E. Seref. Cognitive packet networks. In *Tools with Artificial Intelligence, 1999. Proceedings. 11th IEEE International Conference on*, pages 47 –54, 1999.

[30] A. Ghasemi. *Spectrum sensing in cognitive wireless networks: Requirements, challenges and design trade-offs*. Doctor of philosofy thesis, University of Toronto, Toronto, Canada, 2008.

[31] A. Ghasemi and Y. G. Li. Collaborative spectrum sensing in cognitive radio networks. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.137–143, Baltimore, Nov. 2005.

[32] A. Ghasemi and E. S. Sousa. Collaborative spectrum sensing for opportunistic access in fading environments. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.131–136, Baltimore, Nov. 2005.

[33] A. Ghasemi and E. S. Sousa. Spectrum sensing in cognitive radio networks: Requirements, challenges and design trade-offs. *IEEE Communications Magazine, Feature topic on Cognitive Radio*, vol.46(no.4):pp.32–39, April 2008.

[34] M. Ghozzi, F. Marx, M. Dohler, and J. Palicot. Cyclostationarity-based test for detection of vacant frequency bands. In *Proc. 1st International Conference on Cognitive Radio Oriented Wireless Networks and Communications (CrownCom)*, June 2006.

[35] M. Gudmundson. Correlation model for shadow fading in mobile radio systems. *Electronic Letters*, 27(23):pp.2145–2146, November 1991.

[36] S. Haykin. Cognitive radio: brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, 2(23):pp.201–220, 2005.

[37] IEEE 802 LAN/MAN standards committee. IEEE 802.22 WG on WRANs (wireless regional area networks, 2010. Disponível em: `http://www.ieee802.org/22/`. Acesso em março de 2010.

[38] O. Ileri, D. Samardzija, and N. Mandayam. Demand responsive pricing and competitive spectrum allocation via spectrum server. In *Proc. IEEE DySPAN 2005*, pages pp.194–202, November 2005.

[39] J. Jia, Q. Zhang, and X. Shen. HC-MAC: A hardware-constrained cognitive MAC for efficient spectrum management. *Selected Areas in Communications, IEEE Journal on*, 26(1):106 –117, jan. 2008.

[40] X. Jing, C. Liu, and X. Sun. Artificial cognitive BP-CT ant routing algorithm. In *Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on*, volume 2, pages 1098 – 1103 vol.2, july-4 aug. 2005.

[41] F. K. Jondral. Software-defined radio-basic and evolution to cognitive radio. *EURASIP Journal on Wireless Communication and Networking*, 2005.

[42] H. Kim and K. G. Shin. Efficient discovery of spectrum opportunities with MAC-layer sensing in cognitive radio networks. *IEEE Transactions on Mobile Computing (T-MC)*, 7(5):pp.533–545, May 2008.

[43] H. Kim and K. G. Shin. In-band spectrum sensing in cognitive radio networks: energy detection or feature detection? In *MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking*, pages pp.14–25, New York, NY, USA, 2008. ACM.

[44] S. Krishnamurthy, M. Thoppian, S. Venkatesan, and R. Prakash. Control channel based MAC-layer configuration, routing and situation awareness for cognitive radio networks. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 455 –460 vol.1, oct. 2005.

[45] A. Kumar and K. Shin. Towards context-aware wireless spectrum agility. In *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages pp.318–321, New York, NY, USA, 2007. ACM.

[46] N. Kundargi and A. Tewfik. Sequential pilot sensing of ATSC signals in IEEE 802.22 cognitive radio networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages pp.2789–2792, 31 2008-April 4 2008.

[47] P. Kyasanur. Mesh networking protocols to exploit physical layer capabilities. In *Proc. IEEE Workshop on Wireless Mesh Networks (WiMesh*, 2005.

[48] R. Lent. Linear QoS goals of additive and concave metrics in ad hoc cognitive packet routing. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 36(6):1255 –1260, dec. 2006.

[49] S.-Y. Lien, C.-C. Tseng, and K.-C. Chen. Carrier sensing based multiple access protocols for cognitive radio networks. In *Proceedings of IEEE International Conference on Communications, ICC 2008, Beijing, China, 19-23*, pages 3208–3214, 2008.

[50] Q. Mahmoud. *Cognitive networks: Towards self-aware networks*. John Wiley and Sons, 2007.

[51] D. Maldonado, B. Le, A. Hugine, T. Rondeau, and C. Bostian. Cognitive radio applications to dynamic spectrum allocation: a discussion and an illustrative example. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages 597 –600, nov. 2005.

[52] D. Maldonaldo, B. Le, A. Hugine, T. W. Rondeau, and C. W. Bostian. Cognitive radio applications to dynamic spectrum allocation: a discussion and an illustrative example. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.597–600, Baltimore, Nov. 2005.

[53] J. Mitola. *Cognitive radio: An integrated agent architecture for software defined radio*. Doctor of technology dissertation, Royal Inst. Technol. (KTH), Stockholm, Sweden, 2000.

[54] J. Mitola. *Software radio architecture: Object-oriented approaches to wireless systems engineering*. John Wiley and Sons, 2 edition, 2000.

[55] J. Mitola and G. Q. Maguire. Cognitive radio: Making software radios more personal. *IEEE Pers. Commun.*, 6:pp.13–18, Aug. 1999.

[56] A. Mody, M. Sherman, R. Martinez, R. Reddy, and T. Kiernan. Survey of IEEE standards supporting cognitive radio and dynamic spectrum access. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages pp.1–7, Nov. 2008.

[57] A. Motamedi and A. Bahai. MAC protocol design for spectrum-agile wireless networks: stochastic control approach. In *Proc. of the IEEE DySPAN 2007*, pages pp.448–451, April 2007.

[58] M. A. Oner and F. K. Jondral. Cyclostationarity-based methods for the extraction of the channel allocation information in a spectrum pooling system. In *Proc. IEEE Radio and Wireless Conference*, pages pp.279–282, September 2004.

[59] C. Perkins and E. Royer. Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on*, pages 90 –100, feb 1999.

[60] J. D. Poston, W. D. Horne, M. G. Taylor, and F. Z. Zhu. Ontology-based reasoning for context-aware radios: insights and findings from prototype development. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages pp.634–637, Nov. 2005.

[61] J. Proakis. *Digital communications*. McGraw-Hill, 4th. ed. edition, 2001.

[62] A. Sahai, N. Hoven, and R. Tandra. Some fundamental limits in cognitive radio. In *Allerton Conf. on Comm., Control and Computing 2004*, October 2004.

[63] S. Shellhammer. The spectrum sensing function, April 2007. IEEE 802.22/07-0074r3.

[64] S. Shellhammer and G. Chouinard. Spectrum sensing requirements summary. Technical report, IEEE 802.22, July 2006.

[65] S. Shellhammer, N. Shankar, R. Tandra, and J. Tomcik. Performance of power detector sensors of dtv signals in ieee 802.22 WRANs. In *Proc. of the ACM TAPAS 2006*, August 2006.

[66] A. Sonnenschein and P. M. Fishman. Radiometric detection of spread-spectrum signals in noise. *IEEE Transactions on Aerospace Electronic Systems*, vol.28(no.3):pp.654–660, July 1992.

[67] I. . D. Standard. Ieee p802.22tm/d0.3 draft standard for wireless regional area networks, May 2007. doc. no.22-07-0086-01-0000.

[68] G. Staple and K. Werbach. The end of spectrum scarcity. *IEEE Spectrum*, March 2004. Disponível em: `http://www.spectrum.ieee.org/mar04/3811`. Acesso em março de 2010.

[69] J. Stine. Spectrum management: the killer application of ad hoc and mesh networking. In *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*, pages 184 –193, nov. 2005.

[70] R. Tandra and A. Sahai. Fundamental limits on detection in low SNR under noise uncertainty. In *Proc. International Conference on Wireless Networks, Communications and Mobile Computing*, pages pp.464–469, June 2005.

[71] R. Tandra, A. Sahai, and S. Mishra. What is a spectrum hole and what does it take to recognize one? *Proceedings of the IEEE – special issue on Cognitive Radio*, 97(5):824–848, may 2009.

[72] H. Tang. Some physical layer issues of wide-band cognitive radio system. In *Proc. IEEE 1st Symposium on Dynamic Spectrum Access Networks (DySPAN)*, pages pp.151–159, Baltimore, Nov. 2005.

[73] R. W. Thomas, L. A. DaSilva, and A. B. Mackenzie. Cognitive networks. In *Proceedings of the First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, Baltimore, MD, USA, November 2005.

[74] R. W. Thomas, D. H. Friend, L. A. DaSilva, and A. B. Mackenzie. *Cognitive Radio, Software Defined Radio, and Adaptive Wireless Systems*, chapter Cognitive Networks. Springer, 2007.

[75] Y. Xiao and F. Hu. *Cognitive Radio Networks*. CRC Press, 2008.

[76] B. Zhang, Y. Takizawa, A. Hasagawa, A. Yamaguchi, and S. Obana. Tree-based routing protocol for cognitive wireless access networks. In *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pages 4204 –4208, march 2007.

[77] Z. Zhang and X. Xu. Implementation of cyclic periodogram detection on vee for cognitive radio. In *Global Mobile Congress (GMC'2007)*, pages pp.1–5, Oct 2007.

Capítulo 5

Redes de Sensores Aquáticas

Luiz F. M. Vieira[1], Antonio A. F. Loureiro[1], Antônio O. Fernandes[1], Mario Campos[1]

[1]Departamento de Ciência da Computação - UFMG

Abstract

This tutorial aims at presenting the current state-of-art of underwater sensor networks. It includes an overview of the research in this area, as well as a description of some specific proposals that show current trends and challenges. The area of Underwater Sensor Networks is a recent research topic that has many applications. For instance, environment monitoring, oceanography, and marine biology investigate ocean warming, study interaction between oceans and atmosphere, seismic prediction, pollution detection, monitoring gas fields and oil rings. There are many challenges presented in developing underwater sensor networks. Electromagnetic waves are rapidly absorbed by water, therefore they do not propagate for long distances inside water. Thus, current research uses acoustic communication that suffers from high latency (speed of sound in water is approximately 1500 m/s, five orders of magnitude slower than the speed of light in the vacuum), small bandwidth, high bit error rate, among many other characteristics that difficult communication and demand smart solutions. Furthermore, sensor nodes can move with water currents, allowing a 4D environment monitoring (space and time). However, this mobility increases the difficulty in developing underwater sensor networks.

Resumo

O objetivo deste minicurso é apresentar o estado da arte em redes de sensores aquáticas, bem como um apanhado geral de trabalhos específicos que ilustram as tendências de pesquisa e os principais desafios da área. Rede de sensores aquática é um tema recente de pesquisa que possui inúmeras aplicações. Para citar apenas algumas temos: monitoramento do meio-ambiente, ajuda em estudos de oceanografia, biologia marítima, aquecimento dos oceanos; interação entre oceanos e atmosfera; predição sísmica; detecção de poluentes e substâncias contaminantes; e monitoração de campos de gás e

petróleo. O desenvolvimento de redes de sensores aquáticas possui desafios importantes. Ondas eletromagnéticas não se propagam por longas distâncias na água. Dessa forma, a pesquisa atual utiliza comunicação acústica que é afetada por uma maior latência na propagação de sinal (a velocidade do som na água é aproximadamente 1500 m/s, cinco ordens de magnitude menor que a velocidade da luz no vácuo), uma largura de banda menor, além de uma taxa alta de bits errados, dentre outras características que dificultam a comunicação e exigem soluções inteligentes. Além disso, os nós sensores podem se mover com as correntes marítimas, permitindo um monitoramento 4D (espaço e tempo) do ambiente. No entanto, essa mobilidade aumenta a dificuldade no desenvolvimento das redes de sensores aquáticas.

5.1. Introdução

Por que pesquisar problemas relacionados ao ambiente aquático? O planeta Terra é composto de água. Cerca de dois terços da superfície da Terra é coberta por oceanos, que em grande parte está inexplorada. Além disso, há uma grande quantidade de recursos naturais a serem descobertos. Também, temos o fato de que os oceanos são responsáveis por grande impacto no clima global. E, em algumas situações, como no caso de guerras, os oceanos são um local natural tanto para ataque quanto para defesa. Finalmente, há várias aplicações em potencial para as redes de sensores aquáticas tais como monitoração de exploração de petróleo e manutenção de hidrelétricas [Kong et al. 2005, Vieira 2009].

Essas redes são formadas por nós sensores que possuem capacidade de comunicação. Para monitoração a longo prazo, os nós sensores podem ser montados no fundo do mar e/ou acorrentados a bóias. Para exploração a curto prazo, os nós sensores podem movimentar-se com as correntes marítimas em várias profundidades. Uma forma de implantação dos nós sensores seria jogá-los de avião. A movimentação dos nós sensores permite uma monitoração 4D (espaço e tempo) e uma cobertura de monitoração dinâmica.

A figura 5.1 ilustra um protótipo de nó sensor aquático e uma rede de sensores aquática formada por vários nós sensores que possuem capacidade de comunicação. Ao contrário das redes de sensores terrestres, os nós sensores precisam suportar a pressão da água e, portanto, possuem um encapsulamento mais robustos.

5.1.1. Motivação

Rede de sensores aquática é um tema recente de pesquisa que possui inúmeras aplicações. Algumas áreas e aplicações que podem se beneficiar de Rede de Sensores Aquáticas são [Akyildiz et al. 2005a, Partan et al. 2006, Kong et al. 2005]: oceanografia, biologia marinha, estudos da interação entre oceanos e atmosfera, estudos do clima, aquecimento global, arqueologia no fundo do mar, predições sísmicas, detecção de poluentes e substâncias contaminantes, controle da qualidade da água e exploração e monitoração de campos de gás, óleo e petróleo. É importante observar que cada uma dessas aplicações pode ter características distintas e, consequentemente, levar a diferentes projetos.

5.1.2. Desafios

Ao contrário das redes terrestres, o uso de rádios não é aplicável em redes aquáticas. Ondas eletromagnéticas de alta frequência são rapidamente absorvidas pela água. Uma

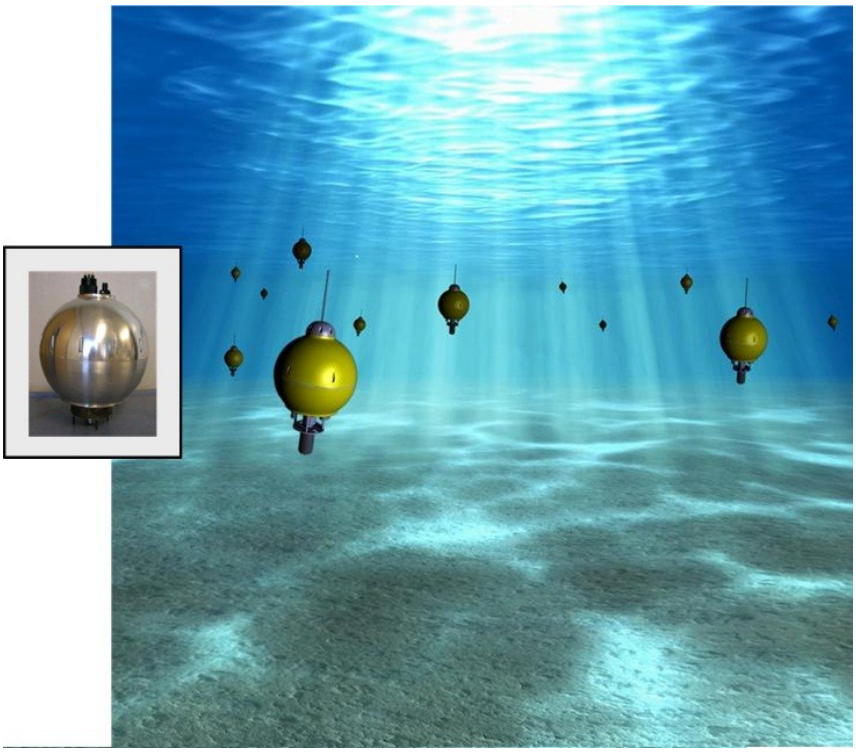


Figura 5.1: Redes de sensores aquáticas.

alternativa seria comunicação ótica, porém a luz tem alcance pequeno, por volta de 10 metros. A solução é utilizar o canal acústico.

O desenvolvimento de redes de sensores aquáticas possui desafios importantes. Ondas eletromagnéticas não se propagam por longas distâncias na água. Dessa forma, a pesquisa atual utiliza comunicação acústica que é afetada por uma latência maior na propagação de sinal (a velocidade do som na água é aproximadamente 1500 m/s, cinco ordens de magnitude menor que a velocidade da luz no vácuo), uma largura de banda menor, além de uma taxa alta de bits errados, dentre outras características que dificultam a comunicação e exigem soluções inteligentes. Além disso, os nós sensores podem se mover com as correntes marítimas, permitindo um monitoramento 4D (espaço e tempo) do ambiente. No entanto, essa mobilidade aumenta a dificuldade no desenvolvimento das redes de sensores aquáticas.

O canal acústico aquático pode ser caracterizado por [Stojanovic 2006]:

- Pequena largura de banda: no máximo centenas de kHz, limitado pela absorção. Atualmente a maior parte dos sistemas acústicos operam abaixo de 30 kHz [Kong et al. 2005];
- Desvanecimento multi-caminho (do inglês *multipath fading*);
- Alta atenuação;
- Largura de banda dependente da frequência e do alcance. Segundo o estudo em

telemetria acústica [Kilfoyle and Baggeroer 2000], o produto banda $\times$ alcance é limitado, em aproximadamente, 40 kbps $\times$ km. Um valor muito baixo quando comparado com rádio em redes terrestres. Por exemplo, o modelo do padrão IEEE 802.11b/a/g gera até 5 Mbps $\times$ km, uma razão de 1 para 100.

- Alta latência. A velocidade do som na água é aproximadamente 1.5×10^3 m/s. A velocidade da luz no vácuo, utilizada na comunicação terrestre, é aproximadamente 3×10^8 m/s. Uma razão de 1 para 10000. Ou seja, cinco ordens de grandeza de diferença.

A velocidade v do som na água é modelada pela equação [Rappaport 1983]:

$$\begin{aligned} v &= 1449.05 + 45.7t - 5.21t^2 + 0.23t^3 + \\ &\quad (1.333 - 0.126t + 0.009t^2)(S-35) + 16.3z + 0.18z^2 \end{aligned} \tag{1}$$

onde t é um décimo da temperatura da água em Celsius, z é a profundidade em metros, e S é a salinidade da água.

Como pode ser observado, a comunicação no meio aquático é bem restrita e isto caracteriza de forma única as redes de sensores aquáticas. Além disso, temos a mobilidade, devido às correntes marítimas, dos nós sensores, por volta de 1 a 1.5 m/s [Kong et al. 2005].

5.1.3. Diferenças para Redes Terrestres

O projeto de redes de sensores aquáticas é significativamente diferente de quaisquer redes terrestre existentes (e.g., redes móveis *ad hoc* e redes de sensores terrestres com e sem fio). Quando comparadas às redes terrestres, as redes de sensores aquáticas apresentam diversas distinções.

A primeira delas é a restrição da largura de banda, bem menor nas redes aquáticas. Também, ao contrário de enlaces sem fio no meio terrestre, cada enlace aquático apresenta alta latência e baixa largura de banda. A latência chega a ser cinco ordens de grandeza maior que nas redes terrestres. Isso significa que um sinal que demora 1 ms na rede terrestre, gasta 100 s na rede aquática. A natureza das aplicações também é diferente. Aplicações para o oceano devem considerar o tamanho das regiões para deposição dos nós sensores. Em geral, as regiões de deposição são vastas e a tendência das redes é de possuírem baixa densidade. A alta taxa de erros presente no meio acústico é maior do que nas redes terrestres, dificultando ainda mais a comunicação e requerendo soluções inteligentes para as aplicações e protocolos.

As principais diferenças entre redes de sensores terrestres e aquáticas são listadas abaixo:

- *Consumo de energia*: A energia necessária para transmitir ondas acústicas é maior que o de ondas de rádios terrestres devido à diferença da tecnologia na camada física. O alto consumo de energia é necessário devido às longas distâncias e ao processamento de sinal mais complexo, que tenta compensar pela natureza de ruídos do canal aquático.

- *Capacidade de memória*: Nós sensores terrestres são equipados com pequena capacidade de armazenamento para manter o baixo preço dos nós sensores. Nós sensores acústicos não precisam seguir essa tendência. Grandes quantidades de armazenamento podem ser instaladas neles, permitindo *caching* de dados para melhor lidar como canal de comunicação tipicamente intermitente.

- *Nível de correlação espacial*: Redes aquáticas são menos densas que as terrestres, devido ao custo mais elevado e o longo alcance dos modems acústicos. Consequentemente, a correlação espacial de informações sensoriadas é relativamente pequena. Isto dificulta a agregação de dados, que é tipicamente realizada em redes de sensores para reduzir o volume do tráfego de dados.

- *Preço do sensores*: Enquanto nós sensors (e.g., motes) em redes de sensores terrestres têm se tornado mais barato com o avanço da tecnologia, nós sensores acústicos usados em redes aquáticas continuam mais caro. O elevado preço se deve à complexidade dos modems aquáticas e da proteção necessária para ambientes aquáticos extremos. Do ponto de vista econômico, o pequeno número de dispositivos em redes aquáticas também é responsável pelo preço elevado.

Essas diferenças são causadas principalmente devido ao fato que redes aquáticas usam ondas acústicas para comunicação, ao contrário das redes terrestres [Akyildiz et al. 2005b, Freitag et al. 2005].

5.1.4. Organização do Texto

A seção 5.2 retrata, em detalhes, o estado da arte, características, particularidades e tendências da área de pesquisa em questão. A seção 5.3 descreve a camada física de redes de sensores aquáticas. A principal diferença das redes de sensores aquáticas para as terrestres é o meio de comunicação, que utiliza o canal acústico. Essa seção também detalha a propagação de sinal e formula a estimativa da entrega de pacotes no meio aquático. A seção 5.4 apresenta os desafios e dificuldades na camada de enlace, bem como os protocolos mais recentes nesse tema de pesquisa. Nessa seção são descritos os protocolos baseados em partição (FDMA, TDMA e CDMA), baseados em acesso aleatório (como Aloha e CSMA) e baseados em reserva do meio e escalonamento (como R-MAC). A seção 5.5 discute protocolos de roteamento: pró-ativos, reativos e geográficos. A seção 5.6 discorre sobre localização. São apresentados métodos recentes de localização: com ajuda de veículos aquáticos, com sinalizadores que ascendem e descendem e, com uso de laser acústico. A seção 5.7 trata serviços de localização. São descritos os métodos baseados em quorum, hashing e ferômonio. A seção 5.8 introduz modelos de mobilidade e descreve o modelo MCM. Também são descritos resultados recentes do uso de MCM em redes de sensores aquáticas, como área cobertura, conectividade e estudo de múltiplas deposições. A seção 5.9 discute algumas aplicações. Finalmente, a seção 5.10 conclui o texto.

5.2. Redes de Sensores Aquáticas: Estado da Arte e Tendências

Esta seção descreve o estado da arte para o caso particular de redes de sensores aquáticas. Atualmente, os oceanógrafos utilizam instrumentos denominados rafos [Rossby et al. 1986] (ver figura 5.2), que são compostos por sensores de profundi-

dade, salinidade e temperatura e são capazes de se movimentar verticalmente, utilizando uma válvula que altera seu volume e, consequentemente, sua densidade. Esses sensores, quando se encontram na superfície, possuem comunicação via satélite. Quando estão no fundo do mar, não há nenhuma comunicação. Não há como saber se os sensores estão funcionando, onde eles se encontram e quais os dados coletados. Em alguns casos, esse período de falta de informação chega a um ano. É desejável que se possa obter os dados e descobrir falhas nos sensores num intervalo de tempo menor que esse. Na tecnologia atual, também existem modems [Freitag et al. 2005] que permitem um enlace de comunicação acústico ponto a ponto.

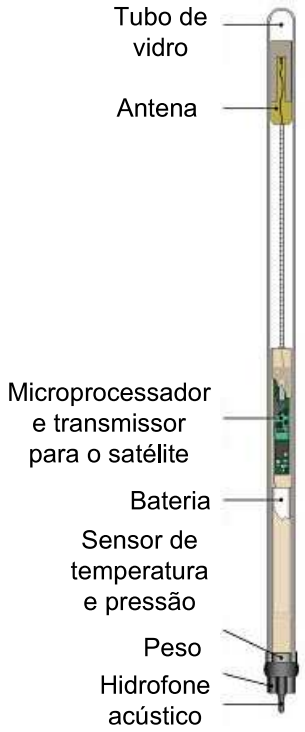


Figura 5.2: Rafos.

Atualmente, grupos de pesquisa ativos na área de redes aquáticas em [WUWNET]: MIT (Prof. Stojanovic), UCLA (Prof. Gerla), USC (Prof. Mitra) e UCSD (Prof. Schurgers). Recentemente [OSU], a Oregon State University, junto com os Institutos Woods Hole Oceanographic (WHOI) e Scripps Institution of Oceanography, tiveram um projeto aprovado, no valor de 386.4 milhões de dólares, para criar um observatório oceânico.

Este tipo de atividade é extremamente importante em um país como o Brasil. E agora, passa a ser mais importante, visto que o governo está decidido a voltar-se a extração de petróleo do fundo do mar. Esse tipo de atividade vai requerer mão-de-obra qualificada e a capacidade de realizar pesquisa de ponta em sensoriamento aquático. Para que se possa usufruir dos recursos naturais presentes na plataforma costeira é necessário que seja possível gerarmos o conhecimento desejável à exploração e defesa de tais recursos.

Acredita-se que, no futuro, teremos redes autônomas para observação dos oceanos, mares, represas, hidrelétricas. Essas redes poderiam ser formadas por rede de

sensores *ad hoc* e/ou frotas de veículos autônomos aquáticos (denominados VAA), em cooperação, como mostrado na figura 5.3.

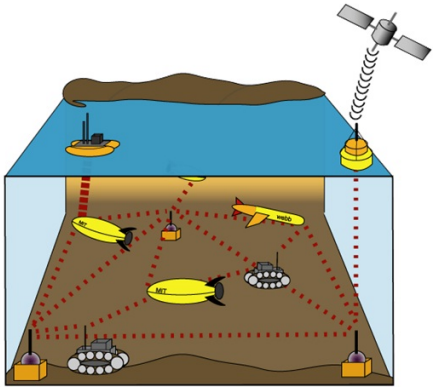


Figura 5.3: Redes autônomas para observação dos oceanos.

5.3. Camada Física e Propagação de Sinal

Esta seção descreve a camada física de redes de sensores aquáticas. A principal diferença das redes de sensores aquáticas para as terrestres é o meio de comunicação. Esse aspecto acaba por influenciar todas as camadas de protocolos, arquitetura do sistema e o desenvolvimento das aplicações.

Ondas eletromagnéticas não se propagam por longas distâncias na água. Dessa forma, a pesquisa atual utiliza comunicação acústica que é afetada por uma latência maior na propagação do sinal. A velocidade do som na água é cinco ordens de magnitude menor que a velocidade da luz no vácuo. Comparada com a comunicação de rádio em redes terrestres, podemos dizer que a largura de banda é menor, a taxa de bits errados é maior, a propagação de sinal sofre com múltiplos caminhos, zonas de sombra, interferência externa, entre outras características que dificultam a comunicação e exigem soluções inteligentes. Descreveremos o modelo de propagação de sinal utilizado atualmente na literatura e empregado em simuladores.

Na camada física, utiliza-se o canal acústico, que possui alta taxa de bits errados, alta latência e baixa largura de banda [Stojanovic 2006].

Tabela 5.1: Largura de banda disponível baseada no alcance de comunicação.

Alcance de Comunicação (km)	Largura de Banda (kHz)
1000	< 1
10–100	2–5
1–10	10
0.1–1	20–50
< 0.1	> 100

5.3.1. Estimativa da Entrega de Pacotes no Meio Aquático

Na literatura, é utilizada o seguinte modelo de canal acústico aquático para estimar a probabilidade de entrega de pacotes [Stojanovic 2006, Brekhovskikh and Lysanov 2003]. A perda na potência do sinal em caminho de distância d em uma frequência f devido ao desvanecimento em larga escala é dada por:

$$A(d,f) = d^k a(f)^d, \tag{2}$$

onde k é o fator de dispersão e $a(f)$ é o coeficiente de absorção. A geometria da propagação é descrita usando um fator de dispersão $(1 \leq k \leq 2)$; para um cenário prático, k é dado por 1.5. O coeficiente de absorção $a(f)$ é descrito pela fórmula de Thorp [Brekhovskikh and Lysanov 2003].

A média da razão entre sinal-ruído (SNR, do inglês *Signal-to-Noise Ratio*) por uma distância d é dada por

$$\Gamma(d) = \frac{E_b/A(d,f)}{N_0} = \frac{E_b}{N_0 d^k a(f)^d}, \tag{3}$$

onde E_b e N_0 são constantes que representam a energia média de transmissão por bit e a densidade de ruído potência em um canal sem desvanecimento e com ruído gaussiano branco aditivo (AWGN, do inglês *non-fading additive white Gaussian noise*). Como mostrado em [Stojanovic 1996, Carbonelli and Mitra 2006], o modelo de desvanecimento Rayleigh é usado para modelar desvanecimento em pequena escala, onde o SNR tem a seguinte distribuição de probabilidade:

$$p_d(X) = \frac{1}{\Gamma(d)} e^{-\frac{X}{\Gamma(d)}}. \tag{4}$$

A probabilidade de erro pode ser avaliada como

$$p_e(d) = \int_0^\infty p_e(X) p_d(X) dX \tag{5}$$

onde $p_e(X)$ é a probabilidade de erro para uma modulação arbitraria para valores específicos do SNR X. Neste texto, descrevemos a modulação BPSK (do inglês *Binary Phase Shift Keying*) que é vastamente utilizada no estado da arte dos modems acústicos [Freitag et al. 2005].

Na modulação BPSK, cada símbolo carrega um bit. A probabilidade de erro no bit, dada um distância d é dada por [Rappaport 2002]:

$$p_e(d) = \frac{1}{2}\left(1 - \sqrt{\frac{\Gamma(d)}{1+\Gamma(d)}}\right) \tag{6}$$

Dessa forma, para qualquer par de nós com distância d, a probabilidade de entrega de pacote com tamanho m bits é simplesmente dada por:

$$p(d,m) = (1 - p_e(d))^m. \tag{7}$$

5.3.2. Utilização do Canal

A utilização do canal η é definida como [Pompili et al. 2006]:

$$\eta = \frac{1}{r}\frac{L_p^d}{T_p \times N^{TX}}, \tag{8}$$

onde r é a taxa de transmissão, N^{TX} é a média do número de transmissões necessárias para o pacote ser recebido e T_p é o tempo total para que o pacote e sua confirmação sejam corretamente recebidos e L_p^d é o tamanho do campo de dados de um pacote L de tamanho L_p.

A figura 5.4, retirada do artigo [Pompili et al. 2006], ilustra a eficiência de utilização do canal de comunicação versus tamanho do pacote (em KB), para diferentes distâncias. A medida que o tamanho do pacote aumenta, aumenta a quantidade de dados transmitidos e isto aumenta a utilização do canal. Porém, ao mesmo tempo, a medida que o tamanho do pacote aumenta, também aumenta a chance do pacote ser recebido com erro, o que cresce o número de retransmissões e diminui a utilização do canal, resultando num ponto ótimo.

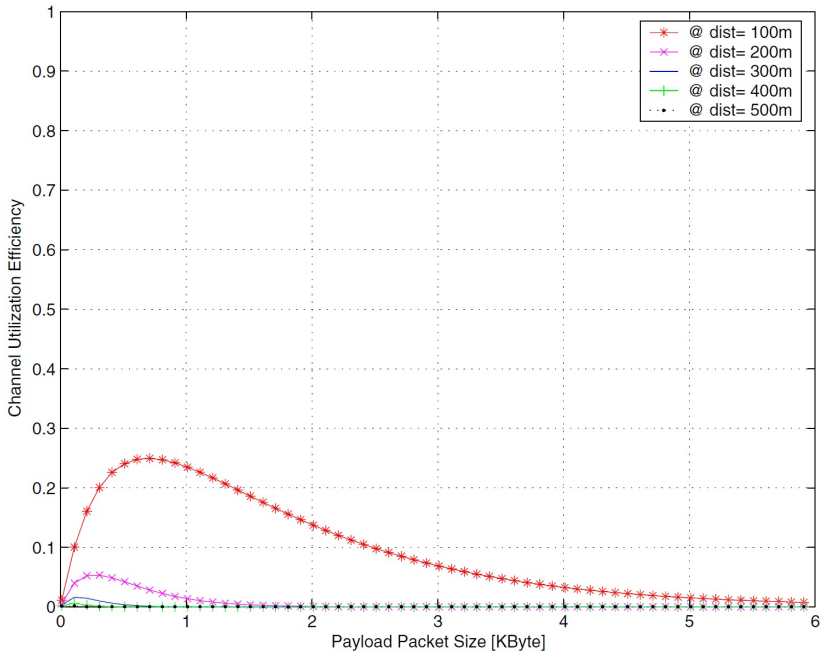


Figura 5.4: Eficiência de utilização do canal vs. tamanho do pacote [Pompili et al. 2006].

5.4. Camada de Enlace

Esta seção descreve os desafios e dificuldades na camada de enlace, bem como apresenta os resultados mais recentes nessa área de pesquisa e os protocolos mais recentes. A seguir, são descritos os protocolos baseados em partição: na frequência (FDMA), no tempo (TDMA), e via código (CDMA). Depois serão descritos os protocolos baseados em acesso aleatório como Aloha e CSMA. Finalmente, serão apresentados os protocolos baseados em reserva do meio e escalonamento.

Protocolos MAC servem para controlar o acesso ao meio comum compartilhado por múltiplos usuários. O objetivo dos protocolos MAC é maximizar a eficiência do e melhorar o *fairness*. Embora protocolos MAC em redes de pacotes de rádio terrestre tenham sido intensamente estudados, eles não podem ser aplicados diretamente às redes de sensores aquáticas devido à dificuldade da natureza do canal de comunicação aquático. Portanto, é imperativo que o desenvolvimento de novos protocolos MAC considerem especificamente as características únicas do meio aquático [Partan et al. 2006, Sozer et al. 2000, Preisig 2007].

Em redes de sensores aquáticas, o espectro disponível é severamente limitado. O problema é fortemente exacerbado devido à velocidade de propagação do som que é cinco ordens de grandeza mais lenta que a baseada em ondas de rádios na superfície terrestre. Dessa forma, esses recursos limitados devem ser compartilhados de forma eficiente e de forma confiável por meios de um protocolo MAC adequado. Nesta seção, os protocolos MAC existentes são categorizados e são discutidas suas vantagens e desvantagens em redes aquáticas. Os protocolos MAC abordados são divididos em três categorias: baseados em partição, acesso aleatório ao meio e baseados em reserva e escalonamento.

5.4.1. Protocolos Baseados em Partição

Na literatura, há três correntes de pesquisa em protocolos baseados em partição: acesso múltiplo dividido pela frequência (FDMA, do inglês *frequency division multiple access*), acesso múltiplo dividido pelo tempo (TDMA, do inglês *time division multiple access*), e acesso múltiplo dividido codificado (CDMA, em inglês *code division multiple access*). Todos os três tipos de protocolos são livres de colisões e o acesso ao canal é justo.

5.4.1.1. FDMA

FDMA é um tipo de protocolo MAC baseado em partição que divide a frequência de banda disponível em múltiplas sub-bandas e assinala cada sub-banda a um nó individual. Em geral, as sub-bandas precisam de bandas de guarda entre elas que compensem filtros imperfeitos, interferência de canais adjacentes e *spectral spreading* devido ao efeito Doppler. A maior vantagem do FDMA é sua simplicidade algorítmica e eficiência quando lidando com pequeno número de nós em um tráfego constante uniforme. No entanto, este esquema não é aplicável em redes aquáticas porque a frequência de banda disponível no canal acústico aquático é pequena.

5.4.1.2. TDMA

Em protocolos TDMA, um intervalo de tempo é dividido em múltiplos períodos de tempo onde cada período é unicamente assinalado a um nó individual. Nesse esquema, os dados são colocados em um buffer até que chegue a vez de seu período de tempo, quando os dados são transmitidos. Isso pode levar a transmissões em rajadas. Infelizmente, transmissões em rajadas necessitam uma taxa alta de transmissão, quando comparados ao FDMA, e aumenta a interferência entre símbolos (ISI, do inglês *Inter Symbol Interference*). Por esse motivo, equalizadores adaptativos são usualmente necessários em

sistemas TDMA [Yackoski and Shen 2008, Kredo et al. 2009].

Como o modem de cada nó sensor é o mesmo, o número de períodos de tempo assinalados a um nó pode mudar sem precisar de hardware adicional. Isso provê ao sistemas TDMA uma flexibilidade. Dessa maneira, a taxa de transmissão dos nós sensores pode aumentar durante o ciclo ocioso [Pahlavan and Levesque]. Da mesma forma, o transmissor pode ser desligado por períodos de descanso, prolongando o tempo de vida da bateria.

Infelizmente, o TDMA requer sincronização de tempo precisa entre todos os nós da rede. Isso implica que todos os nós devem ter mecanismos instantâneos de comunicação para permanecerem sincronizados durante falhas dos nós ou movimento dos nós para manter o escalonamento de transmissões. Entretanto, em comunicação acústica, devido à grande latência na propagação do sinal, cada período de tempo deve ser mantido longo o suficiente para evitar colisões. Isso pode levar a longos períodos de tempo sem comunicação, prejudicando severamente a vazão.

5.4.1.3. CDMA

Protocolos CDMA permitem que múltiplos nós operem concorrentemente sobre toda a frequência de banda. O acesso ao canal de nós diferentes é baseado em um código único que é usado para espalhamento. Existem duas técnicas básicas de espalhamento, DSSS (do inglês *direct sequence spread spectrum*) e FHSS (do inglês *frequency hopped spread spectrum*).

No caso do DSSS, sinais de informação são linearmente modulados usando códigos de banda larga, enquanto no FHSS as frequências portadoras dos nós são modificados de acordo com um padrão obtido a partir dos códigos [Pahlavan and Levesque]. Um único código é assinalado a cada nó para modulação. Enquanto o CDMA é robusto ao desvanecimento da frequencia selecionada, beneficia de múltiplas recepções simultâneas, e pode compensar pelo efeito de múltiplos caminhos através de filtros Rake [Pompili et al. 2009]. Porém o CDMA é vulnerável ao problema perto–longe[1] (também conhecido como *near-far problem*) [Flikkema]. Adicionalmente, um algoritmo de controle de potência é necessário para reduzir o nível da potência de saída de cada nó para estabelecer transferência de pacotes confiável sem criar interferência excessiva e para aliviar o consumo excessivo de bateria em transmissões aquáticas.

Pompili *et al.* [Pompili et al. 2009] propôs um algoritmo distribuído para um protocolo CDMA que obtém a transmissão ótima de potência e tamanho do código. Resultados de simulação mostram que o protocolo apresenta uma melhora em relação a protocolos MAC existentes sintonizados para águas rasas, em diferentes cenários de simulação. Entretanto, vários problemas precisão ser resolvidos, como separação de sinal e a atribuição dos códigos. De acordo com os autores, CDMA e sinalização de espalha-

[1]Considere um receptor e dois transmissores, um perto e outro longe do receptor. Se ambos os transmissores transmitirem numa potência igual, então devido ao decaimento proporcional a distância, o receptor irá receber mais potência no sinal do transmissor mais perto. Como o sinal de transmissão é ruído para o outro, a razão sinal-ruído (SNR) para o transmissor mais distante será mais baixa. Isso faz com que a transmissão do mais distante fique praticamente impossível de ser entendida.

mento de espectro são técnicas promissoras para múltiplos acessos em redes em águas rasas (profundidade menor que 100 m).

5.4.2. Protocolos Baseados em Acesso Aleatório

Estes tipos de protocolos são baseados em acesso aleatório. Este esquema não divide os recursos limitados do canal, mas permitem que nós acessem o meio baseado no princípio da contenção. Assim, esses tipos de protocolos aumentam a utilização do canal, mas não garantem acesso livre de colisão entre os nós.

5.4.2.1. ALOHA

O protocolo Aloha original é baseado em puro acesso aleatório ao meio. Quando um nó tem informação a enviar, ele transmite a informação imediatamente. Uma confirmação (ACK) é enviada de volta pelo receptor se o pacote for recebido corretamente. Se a colisão ocorrer, devido à transmissão concorrente de dados ou ACKs no receptor e transmissor respectivamente, o transmissor retransmite o mesmo pacote. Devido à contenção e retransmissões, a vazão máxima alcançável pelo Aloha original é de 18% [Bertsekas and Gallager , Vieira et al. 2006].

Roverts *et al.* [Roberts 1975] propuseram dois protocolos distribuídos baseados no Aloha para comunicação aquática: Aloha com *Collision Avoidance* (Aloha-CA) e Aloha com *Advance Notification* (Aloha-AN). Nesses dois protocolos, cada nó tenta utilizar a informação sobre o transmissor e o receptor, escutando outras transmissões, para ajudar a evitar colisões e obter um desempenho de vazão melhor.

No Aloha-CA, cada pacote é diferenciado em dois segmentos distintos, um cabeçalho e um segmento de dados. Baseado na informação de transmissor-receptor obtida escutando a transmissão, cada nó calcula a duração que o meio estará ocupado da transmissão em andamento e decide se esse estado ocupado é causado pela transmissão, recepção ou escuta de pacote. Essa decisão ajuda cada nó a aumentar a vazão e reduzir a chance de colisão de pacotes. Esse esquema não requer sincronização porque cada nó mantém informação localmente na sua tabela de dados com respeito ao seu próprio relógio. Reduzindo o tamanho do cabeçalho, o protocolo Aloha-CA reduz o tempo necessário para obter informações úteis e diminui a possibilidade de colisão de pacotes. Entretanto, colisões ainda são possíveis porque a tabela é mantida somente com a informação que o nó escutou anteriormente, que é meramente um subconjunto de toda a rede e que é necessário para decisões livres de colisões.

O protocolo Aloha-AN modifica o Aloha-CA. Cada nó primeiro transmite seu segmento de cabeçalho como um pequeno pacote de notificação avançada (NTF). O transmissor irá esperar por um período de tempo, denominado *lag time*, antes de enviar um pacote de dados. Cada nó na rede, que escuta o NTF, checa se o pacote de dados associado irá causar conflito com seu próprio pacote de dados escalonado para transmitir ou receber, e calcula o tempo esperado de transmissão. Se existe qualquer conflito com sua transmissão ou com nós vizinhos, o mecanismo de resolução é invocado. Se um conflito é detectado entre dois nós, o mecanismo escolhe o nó com o escalonamento mais cedo para transmitir primeiro e o outro aguarda. O nó que aguarda, tem que transmitir um novo

pacote NTF.

A diferença chave entre os dois mecanismo é que o Aloha-AN já considera a transmissão de dados escalonada obtida via NTF quando escalonando a própria transmissão. Se um novo escalonamento entra em conflito com o escalonamento existente, o escalonamento pode ser posto em espera. Portanto, quando comparado com o Aloha-CA, o NTF dá ao nó um subconjunto maior do estado da rede, permitindo tomar uma decisão melhor para evitar colisões.

O custo adicional inserido na transmissão do pequeno pacote NTF é justificado pelos casos de colisão de pacotes que iriam ocorrer caso o mecanismo de informação adicional não existisse. Esse mecanismo não é totalmente livre de colisão. Ele também não garante justiça, visto que um nó pode continuamente enviar pacotes NTF para capturar o canal indefinidamente, enquanto tiver dados para transmitir. Finalmente, esse algoritmo não pode realizar ciclos ociosos para economizar energia porque a informação adicional é coletada escutando o canal, requerendo que o nó permaneça ligado escutando os pacotes NTF.

5.4.2.2. CSMA

Recursos limitados do canal podem ser melhor utilizados se os nós sensoriam as portadoras antes de transmitir. O método de acesso ao meio é baseado nesta idéia e é chamado CSMA (*carrier sense multiple access*) [Pahlavan and Levesque]. O método CSMA tenta evitar colisões ao escutar o meio na vizinhança do transmissor. Nesta abordagem, apenas colisões no lado do transmissor são tratadas, o receptor ainda pode sofrer colisões devido ao problema do terminal escondido [Bharghavan et al. 1994]. Detalhes e variantes desse método podem ser encontrados em [Smith et al. 1997, Kleinrock and Tobagi 1975, Tobagi and Kleinrock 1975, Takagi and Kleinrock 1987].

O protocolo MACA proposto por Karn [Karn 1990] para detectar colisões no receptor é uma alternativa ao CSMA. Este protocolo introduz troca de mensagens ao adicionar dois pacotes de controle chamados RTS (*request-to-send*) e CTS (*clear-to-send*). Quando um nó quer enviar mensagem a outro, primeiro ele envia um RTS que contém o tamanho da mensagem que será enviada. Se o destinatário receber o RTS, ele responde com um CTS que também tem o comprimento da mensagem. Logo que o transmissor recebe o CTS, ele inicia a transmissão de dados. Qualquer nó que escuta o CTS deixa de transmitir pelo tempo necessário para evitar colisão com o pacote sendo transmitido. Se um nó vizinho escuta o RTS mas não o CTS, ele infere que está fora do alcance do receptor e transmite o próprio pacote. Este protocolo confia na simetria do canal; o CTS deve ser escutado por todos os nós dentro do alcance do nó receptor. Isso implica que o controle de potência não pode ser usado com estes tipos de protocolos.

Fullmer *et al.* [Fullmer and Garcia-Luna-Aceves 1995] propôs FAMA, que estende a duração dos pacotes de controle RTS (maior que o máximo de atraso na propagação) e CTS (comprimento do RTS mais duas vezes o máximo da propagação, mais tempo de transição do hardware para transmitir/receber) para prevenir alguns tipos de colisões. Para adaptar este protocolo para redes aquáticas, Molins *et al.* [Molins 2006] propôs o protocolo Slotted FAMA. O princípio do algoritmo Slotted FAMA é baseado

no algoritmo FAMA. A diferença primária é que o canal é divido em períodos de tempo e pacotes (RTS, CTS, DATA or ACK), que são enviados apenas por nós no começo de um intervalo de tempo. Se um nó quer enviar um pacote, ele tem que esperar o começo do próximo intervalo de tempo para iniciar o algoritmo FAMA. Esta técnica de divisão do tempo em intervalos é provada [Roberts 1975] que minimiza as chances de colisão e, portanto, aumenta a vazão. Ao contrário do FAMA, o receptor do pacote pode imediatamente encaminhar o pacote sem ter que esperar um tempo de contenção, permitindo o slotted FAMA alcançar uma melhor eficiência.

No entanto, como mencionado no caso do TDMA, sincronização precisa entre todos os nós da rede é difícil de ser implementada em redes aquáticas. Além disso, esse protocolo tem bom desempenho apenas com um pequeno número de nós, visto que a vazão diminui com o aumento do número de nós.

5.4.3. Protocolos Baseados em Reserva e Escalonamento

Protocolos baseados em reserva e escalonamento utilizam um acesso via escalonamento determinístico para maximizar a utilização do canal [N. Chirdchoo, W.-S. Soh, and K.-C. Chua 2008]. Esse esquema não divide os recursos entre os nós, mas permite que eles acessem o canal via um escalonamento pré-definido. Portanto, enquanto esses tipos de protocolos aumentam a utilização do canal e transmissões sem colisões, eles não aceitam mudanças dinâmicas na rede, como nó juntar a rede, sair, falhar ou mover.

5.4.3.1. R-MAC: Procotolo MAC Baseado em Reserva

R-MAC é um protocolo tipo CSMA baseado em escalonamento [Xie and Cui 2007]. Ele alcança eficiência de energia usando escuta periódica e modos de dormir para reduzir a energia gasta em estados ociosos. Todos os nós usando este protocolo têm a mesma duração de períodos para escutar e dormir, mas aleatoriamente selecionam seu escalonamento de tal forma a ter eficiência de energia e justiça de acesso ao meio.

R-MAC compreende três fases: detecção de latência, anúncio do período, e operação periódica. As duas primeiras fases sincronizam cada nó com seus vizinhos e a terceira realiza operações de escutar/dormir e comunicação de dados. Um nó na fase de detecção de latência detecta a latência de propagação para todos os seus nós vizinhos. Na fase de anúncio do período, cada nó aleatoriamente escolhe seu próprio escalonamento para escutar/dormir e propaga essa informação. Os dados são transmitidos na fase periódica de operação.

Durante a transmissão de dados, os nós comunicam através de mensagens dos tipos REV/ACK-REV/DATA/ACK-DATA. Quando um nó possui dados para transmitir, primeiro ele envia um REV (pacote de reserva) para fazer a reserva do intervalo de tempo no receptor. Uma vez que o pretendido receptor está pronto para receber os dados, ele notifica seus vizinhos, assim como o transmissor, enviando uma mensagem do tipo ACK-REV, reconhecendo que está pronto para receber. Ao receber o pacote ACK-REV, todos os outros, com exceção do transmissor, ficam em silêncio, enquanto o transmissor envia dados no intervalo de tempo. O receptor envia de volta uma mensagem ACK-DATA para

o transmissor no fim da transmissão para liberar o meio.

Este mecanismo de escalonamento baseado em reserva permite aos nós dormirem, economizando energia e resolvendo o problema do terminal escondido presente em protocolos baseados em RTS/CTS. Entretanto, este protocolo falha em topologias dinâmicas e móveis. Embora este protocolo seja baseado em escalonamento, desvios no relógio do sistema não são considerados no trabalho original.

5.5. Roteamento

Pesquisa em roteamento para rede móveis é uma área ativa que possui vários protocolos desenvolvidos. Nesta seção são descritas as peculiaridades dos protocolos de roteamento para redes de sensores aquáticas. Também são descritos os desafios em utilizarmos protocolos desenvolvidos inicialmente para redes terrestres, tanto pró-ativos quanto reativos, como, por exemplo, OLSR e AODV.

Protocolos de roteamento desenvolvidos para redes móveis terrestres não possuem um bom desempenho. Em geral, eles podem ser classificados em três tipos: pró-ativo (e.g., OLSR [OLS]), reativos (e.g., AODV [Perkins 1997]) ou geográfico. Protocolos pró-ativos e reativos requerem a descoberta de rota (através de *flooding*) e/ou manutenção das rotas. Portanto, não são apropriados para comunicações com largura de banda restrita. Além disso, também há o problema de colisões de mensagens e alto consumo de energia. Dessa forma, o roteamento geográfico [Füßler et al. 2003, Kalosha et al. 2008, Flury and Wattenhofer 2008] é preferível, porém este requer um serviço de localização que provê a localização do destinatário.

Protocolos pró-ativos (e.g., DSDV [Perkins and Bhagwat 1994], OLSR [Clausen and Jacquet 2003]) possuem um grande custo de sinalização para estabelecer rotas pela primeira vez e para cada vez que a topologia da rede é modificada por causa de mobilidade ou falha dos nós. A informação de atualização da topologia da rede deve ser propagada por todos os dispositivos da rede. Dessa forma, todo nó da rede é capaz de estabelecer um caminho para qualquer outro nó na rede, o que pode não ser preciso em redes aquáticas. Por esta razão, protocolos pró-ativos não são adequados para redes aquáticas.

Protocolos reativos (e.g., AODV [Perkins 1997], DSR [Johnson et al.]) são mais apropriados para ambientes dinâmicos mas incorrem em maior latência e ainda requerem um *flooding* iniciado pela fonte ou pacotes de controle para estabelecer rotas. Protocolos reativos não são adequados para redes aquáticas porque eles causam alta latência no estabelecimento de rotas, o que aumenta ainda mais o efeito da propagação lenta do meio acústico na água.

Protocolos de roteamento geográfico (e.g., GFG [Bose et al. 1999], GPSR [Karp and Kung 2000]) são muito promissores pela sua característica de escalabilidade e pequena quantidade de sinalização necessária. Entretanto, receptores de GPS (*Global Positioning System*), que podem ser usados em sistemas terrestres para estimar com precisão a localização geográfica dos nós sensores, não funcionam no meio aquático. O fato é que o GPS usa ondas de 1.5 GHz e essas ondas não propagam na água. Mesmo assim, dispositivos nas redes de sensores aquáticas (sensores, veículos

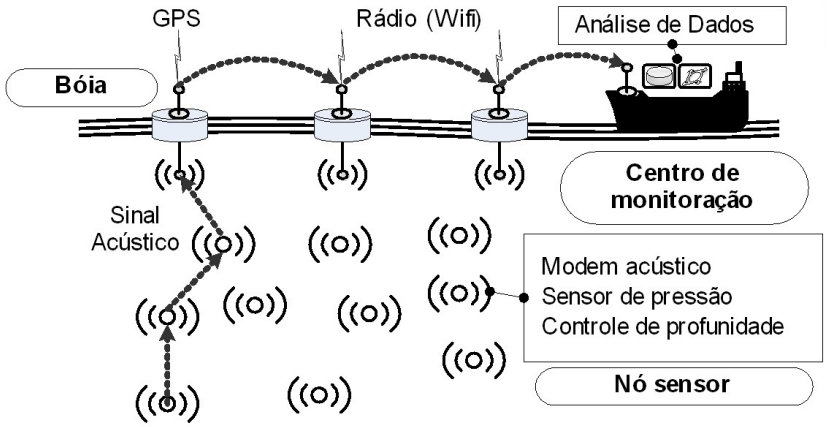


Figura 5.5: Roteamento baseado na pressão.

autônomos, etc.) precisam estimar sua posição atual, independente do mecanismo de roteamento. Na realidade, já é necessário associar os dados coletados com a posição 3D dos dispositivos que coletaram os dados, para reconstruir espacialmente as características dos eventos. Portanto, localização é uma tarefa necessária nas redes de sensores aquáticas e sua presença facilitaria os protocolos de roteamento através de algoritmos geográficos.

Recentemente, foi proposto um protocolo para redes aquáticas baseado na pressão [Lee et al. 2010]. O cenário de aplicação é bem especializado já que qualquer bóia na superfície pode receber os dados, o que permite resolver o problema por um *anycast* geográfico. Dessa forma, é suficiente encaminhar o pacote para cima até atingir a superfície, como mostrado na figura 5.5. Dado que um sensor de pressão pode estimar a profundidade com razoável precisão (erro médio < 1 m [Jalving 1999]), a informação de profundidade pode ser usada para roteamento geográfico *anycast* como discutido em [Yan et al. 2008]. Decisões de encaminhamento de pacotes são feitas localmente tendo como base a medida do nível de pressão (ou profundidade) em cada nó sensor. O pacote é encaminhado de forma gulosa para os vizinhos com menor pressão.

5.6. Localização

Esta seção descreve os desafios e métodos desenvolvidos para localização dos nós sensores subaquáticos. Localização é um problema importante que possui inúmeras dificuldades. Por exemplo, o sinal de GPS não propaga debaixo da água. Localização é necessária para se identificar o local dos dados coletados e determinar a posição dos eventos ocorridos na rede. Também é essencial em algoritmos de roteamento geográfico. Serão descritos métodos recentes de localização baseados em veículos autônomos, *beacons* e laser acústico.

Localização é essencial para identificar o dado sensoriado. O dado coletado na rede tem pouca utilidade sem a informação de onde ele foi obtido. Localização é necessária em várias funções de alto nível da rede, como monitoração, rastreamento e roteamento baseado em posição geométrica. O ambiente aquático é ainda mais restrito. Um receptor de GPS, utilizado para localização em uma rede terrestre de ambiente aberto, não pode ser usado debaixo da água porque não há propagação desse sinal. Além disso, redes aquá-

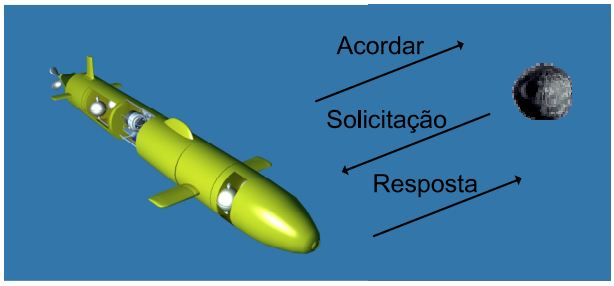


Figura 5.6: Localização com Ajuda de Veículos Autônomos.

ticas exigem que soluções para o problema de localização utilizem um número pequeno de troca de mensagens devido ao consumo de energia e baixa largura de banda presente no meio acústico de comunicação.

Portanto, é necessário um método eficiente em termos de energia e número de mensagens para a localização de nós sensores em redes de sensores aquáticas. Uma solução proposta é a utilização de veículos autônomos aquáticos, e outra é usar sinalizadores especiais denominados DNR (do inglês *dive and rise*) [Erol et al. 2007a, Erol et al. 2007b] que aprendem a sua posição enquanto estão na superfície da água. Um outro método descrito utiliza laser acústicos [Vieira et al. 2009].

5.6.1. Localização com Ajuda de Veículos Autônomos

Erol *et al.* [Erol et al. 2007a] propõem utilizar veículos aquáticos autônomos (VAA) para ajudar a localizar os nós sensores. A figura 5.6 ilustra esse sistema. Em algumas aplicações, como emergências, os nós sensores podem ser jogados na água e permanecer lá por alguns dias. Como os nós não estão fixos, eles se movem com as correntes marítimas. O VAA pode ajudar a estabelecer uma atualização periódica da localização dos nós. O VAA anuncia sua presença enviando um sinal para os nós acordarem. Os dispositivos que receberem o sinal irão iniciar o processo de localização enviando um pacote de solicitação. O VAA responde com um pacote que inclui a sua coordenada. O atraso na propagação entre os dois pacotes é utilizado para calcular a distância (assumindo velocidade uniforme do som na água). Existem várias maneiras de estimar as coordenadas usando medições de alcance.

No método proposto, é utilizada a caixa delimitadora (do inglês *bounding box* [Savvides et al. 2002]). Esse método precisa de duas mensagens enviadas de duas posições não alinhadas para desenhar uma região retangular com as distâncias estimadas. As coordenadas de um nó são dadas pela interseção das diagonais do retângulo. O desempenho do método é dependente da posição das âncoras. Um nó é melhor localizado se receber sinais de lados opostos da caixa.

Localização com ajuda de veículos autônomos usa três mensagens: acordar, solicitar e responder. O VAA envia a mensagem de acordar (4 bytes) para notificar sua presença aos nós dentro do alcance de comunicação. Neste caso, assume-se que os nós não estão sincronizados e, portanto, utiliza-se uma mensagem de solicitação (4 bytes) e de resposta (24 bytes) para medir o tempo de propagação (RTPD, do inglês *Round Trip Propagation Delay*). Assumindo velocidade uniforme e homogênea do som, estima-se a distância via

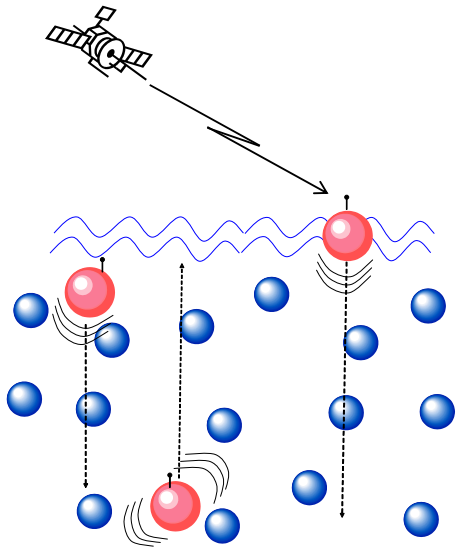


Figura 5.7: Localização via *Dive and Rise*.

(velocidade do som $\times$ *RTPD/2)*. A mensagem de resposta inclui a coordenada do VAA.

5.6.2. Localização DNR (*Dive and Rise*)

Em operações a longo prazo, a solução comum para localização tem sido usar a técnica LBL que é baseada na colocação de sinalizadores na superfície. Nessas operações, usualmente os sensores são colocados distantes um do outro e eles coletam dados individualmente. Eles transferem os dados coletados para uma estação central via enlace de satélite. Eles não comunicam um ao outro, em outras palavras, eles não formam uma rede. Entretanto, aplicações atuais de oceanografia requerem rede. Como a necessidade de redes surge, para atingir altas taxas de transferência, há a redução da distância entre os nós sensores. Para a localização em uma rede desse tipo, os sinalizadores devem ser substituídos por alternativas de pequeno alcance. Nesse caso, a informação de localização tem que ser iterativamente encaminhada para os nós que não estão no raio de transmissão dos sinalizadores na superfície ou algum dispositivo móvel precisa entregar as coordenadas no estilo GPS, se movendo na região próxima aos nós. Para estender a informação de localização global do serviço GPS para debaixo da água, Erol *et al.* [Erol et al. 2007b] propõem a técnica DNR, mostrada na figura 5.7.

DNR usa sinalizadores móveis para distribuir as coordenadas no estilo GPS para os nós sensores das redes aquáticas. Os sinalizadores DNR aprendem suas coordenadas enquanto flutuam na superfícies do oceano. Depois disso eles periodicamente submergem aos níveis mais profundos da rede e sobem de volta para receber informações de sua localização corrente. Enquanto submergem e sobem, os sinalizadores DNR enviam mensagens com sua localização. Os nós sensores passivamente escutam a essas mensagens e, portanto, eles não gastam muita energia no processo de localização.

A mensagem de localização inclui um campo de *timestamp* e as coordenadas das três dimensões do sinalizador DNR. O campo de *timestamp* é usado para calcular a distância entre o sinalizador e os nós, através da técnica *ToA* (do inglês *time of arrival*). É

apropriado assumir que a diferença entre o tempo de chegada e o *timestamp* multiplicada pela velocidade do som na água resulta na distância entre o sinalizador e o nó sensor. Assume-se que os nós estão sincronizados e que a velocidade do som na água é constante na região de transmissão. Os nós podem assumir que estão sincronizados por várias semanas depois da deposição inicial. Entretanto, se a rede for utilizada para uma missão de longo prazo, é claro que um protocolo adicional de sincronização é necessário para ser executado antes da localização.

No método DNR, quando um nó recebe mensagens de três ou mais sinalizadores, ele pode calcular as suas coordenadas via triangulação. O método em si pode ser usado para estimar n coordenadas se existirem $n+1$ ou mais mensagens dos sinalizadores. O método é baseado na idéia de interseção de círculos. É um método bastante difundido que também é utilizado pelo sistema GPS. As coordenadas estimadas devem satisfazer o conjunto de equações:

$$(x-x_i)^2+(y-y_i)^2+(z-z_i)^2=d_i^2 \tag{9}$$

onde i denota o identificador do sinalizador, (x_i,y_i,z_i) são as coordenadas dos sinalizadores e d_i é a distância medida entre o sinalizador e o nó.

Note que três equações independentes são suficientes para resolver este sistema não linear de equações para (x,y). Como os nós sensores têm sensores de pressão, a informação de profundidade, i.e., coordenada z já é conhecida. O sistema de equações é linearizado via subtração da $n+1^{th}$ equação das primeiras n equações. Depois disso, as coordenadas são estimadas com o método do menor quadrado. Aqui, resolve-se $A\varphi=b$ onde,

$$A=\begin{bmatrix} 2(x_1-x_n) & 2(y_1-y_n) \\ . & . \\ . & . \\ . & . \\ 2(x_{n-1}-x_n) & 2(y_{n-1}-x_n) \end{bmatrix}$$

$$b=\begin{bmatrix} x_1^2-x_n^2+y_1^2-y_n^2+z_1^2-z_n^2-2z(z_1-z_n)+d_n^2-d_1^2 \\ . \\ . \\ . \\ x_{n-1}^2-x_n^2+y_{n-1}^2-y_n^2+z_{n-1}^2-z_n^2-2z(z_{n-1}-z_n)+d_n^2-d_{n-1}^2 \end{bmatrix}$$

As coordenadas $\hat{\varphi}=[\hat{x}\ \hat{y}]^T$ são estimadas usando o método de menor quadrado: $\hat{\varphi}=(A^TA)^{-1}A^Tb$.

No método DNR, um nó é considerado localizado se a estimativa do erro for menor que o alcance de comunicação, R. O erro, ε, é definido como a diferença entre a distância estimada e a distância medida [Langendoen and Reijers 2003]. A distância estimada é a distância resultante do uso das coordenadas estimadas e das coordenadas dos sinalizadores. A distância medida é calculada via ToA.

$$\varepsilon=\frac{1}{n}\sum_{i=1}^{n}\sqrt{(x_i-\hat{x})^2+(y_i-\hat{y})^2+(z_i-z)^2}-d_i. \tag{10}$$

Se $\varepsilon > R$ então o nó é marcado como não-localizado. Como a localização é feita periodicamente, um nó não-localizado pode se tornar localizado mais tarde, e um nó localizado pode refinar sua estimativa.

5.6.3. LPS (*Laser Positioning System*)

Esquemas alternativos têm sido propostos para localização em redes aquáticas. Nessas redes, o limite de energia nas baterias é crítico. Protocolos devem economizar trocas de mensagens e energia. Métodos de posicionamento frequentemente consomem energia porque requerem trocas de mensagens periodicamente. Recentemente, a tecnologia de laser acústico avançou. Vieira *et al.* [Vieira et al. 2009] propôs o LPS (*Laser Positioning System*) para redes de sensores aquáticas.

Experimentos recentes [Jones et al. 2006] demonstraram que um pulso de laser pode propagar na água e no final explodir gerando um pulso acústico a uma distância pré-determinada da superfície. Jones *et al.* [Jones et al. 2006] mediu a duração de um pulso na ordem de 1 μs e níveis de pressão sonora de até 170 dB. A fonte de laser pode estar acima da água, por exemplo, em um avião. Os autores propõem usar este pulso acústico gerado pelo laser como um sinalizador de localização. Escutando o pulso acústico, e sabendo o tempo e a localização exata, os nós sensores podem se localizar via triangulação.

Os benéficos do LPS são:

- Simplicidade: virtualmente equivalem a um GPS para ambiente aquático.
- Operação Secreta: o laser não pode ser detectado por espiões ou inimigos acima da água; os nós sensores são totalmente passíveis e podem preservar a privacidade.
- Mensagens são encriptadas; assim, apenas membros podem recuperar os valores da posição e tempo dos pulsos acústicos.
- Fácil deposição, não requer submarinos, navios, instalações ou estudos de deposição.
- Eficiente em energia; os nós sensores não precisam transmitir nada.

A figura 5.8 ilustra a arquitetura do Sistema de Posição via Laser. Qualquer dispositivo acima da água, por exemplo, um avião, pode emitir o laser. Os nós sensores aquáticos formam um enxame flutuando com as correntes marítimas. O avião sobrevoa a área do enxame de sensores, por exemplo, a 1000 metros. Ele tem cinco lasers: quatro apontam para vértices de um quadrado na água, e o quinto aponta para o centro da região.

O avião primeiro atira quatro pulsos de lasers (e.g., 1, 2, 3, 4). Esses lasers atingem a água e geram um pulso acústico a uma profundidade fixa. Resultados recentes em laser acústico reportam valores típicos de 10 metros de profundidade [Jones et al. 2006]. O pulso acústico é gerado a intervalos pré-definidos (e.g., um segundo) de tal forma que eles possam ser recebidos distintamente por todos os nós que estão no alcance de comunicação. Logo depois, o avião atira o quinto laser, que atinge a água perpendicularmente a aeronave.

Usando o método de modulação 4-MFSK, pesquisadores demonstraram que é possível alcançar uma taxa de transmissão de dados de 160 bits/s por 320 metros num enlace aquático [Blackmon and Antonelli 2006]. O quinto laser pulsa de forma a transmitir um pacote que contém o tempo e as coordenadas dos quatro pontos de impacto dos pulsos de lasers anteriores (tempo, x, y). Não há necessidade para a coordenada z porque ela é aprendida pelos sensores via pressão. Esse pulso sinalizador é criptografado para não dar coordenadas para o inimigo. Ele

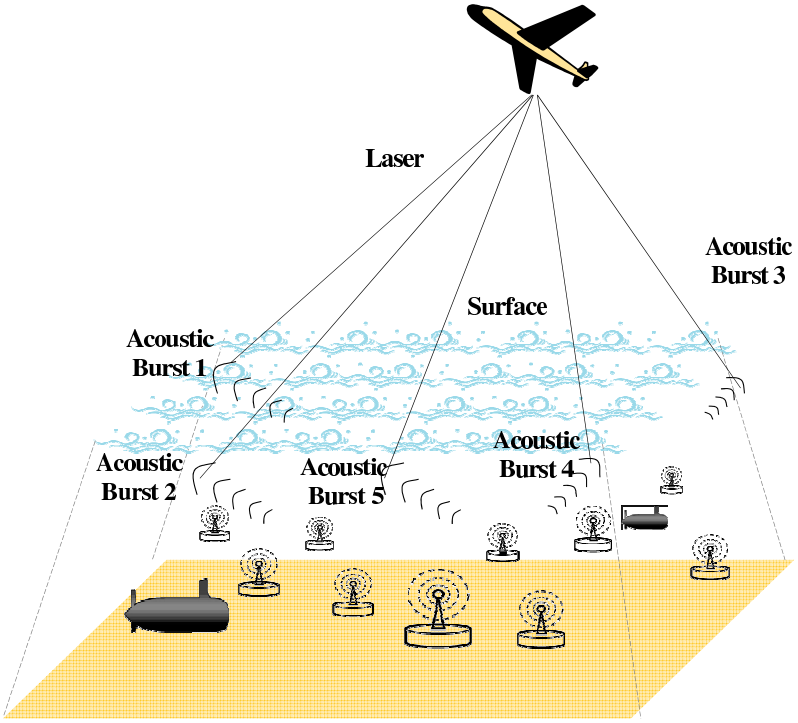


Figura 5.8: Laser Position System Architecture.

também é autenticado, evitando ataques maliciosos. Somente nós sensores que são membros do enxame recuperam os valores de posição e tempo.

Quando o feixe de laser atinge a água, ele cria um pulso acústico que vai até o fundo do mar. Cada nó sensor, ao receber o pacote, computa sua distância para o ponto de impacto. Após escutar três pulsos, os sensores se localizam passivamente (sem a necessidade de enviar mensagens) usando triangulação.

5.7. Serviço de Localização

Um serviço de localização consiste em prover, para qualquer nó da rede, a localização de um nó. As principais operações são atualizar a localização de um nó e consultar a localização de outro nó sensor.

O serviço de localização é um pré-requisito em redes que utilizam um algoritmo geográfico porque o nó fonte precisa saber a localização do destinatário quando for enviar a mensagem. Através do serviço de localização, o nó destinatário fica atualizando a sua localização. Quando um nó fonte deseja enviar uma mensagem ao nó destinatário, primeiro ele realiza uma consulta ao serviço de localização para determinar a posição do destinatário e utilizar um dos algoritmos geográficos existentes para encaminhar a mensagem até o destino final. Serão descritos métodos baseados em quorum (um subconjunto de nós), hashing e métodos inspirados na biologia (feromônio).

5.7.1. Métodos Baseados em Quorum

Em métodos baseados em quorum [Haas and Liang 1999], cada atualização é enviada para um conjunto de nós (quorum de atualização); similarmente, uma consulta de localização é enviada

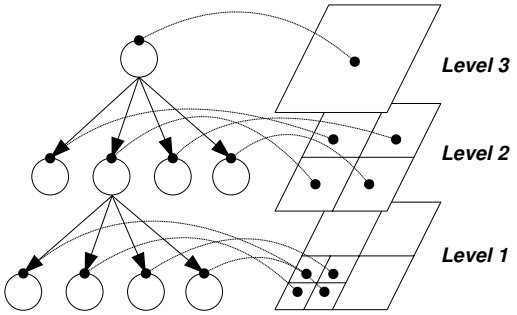


Figura 5.9: Esquema hierárquico.

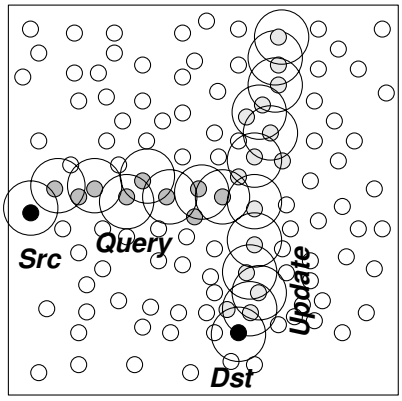


Figura 5.10: Esquema baseado em quorum: XYLS

a outro conjunto de nós (quorum de consulta). Se os dois conjuntos (atualização e consulta) se interceptam, a consulta é resolvida. Vários métodos para manter o quorum têm sido propostos [Friedman and Kliot 2006].

No XYLS [Stojmenovic et al. 2006], um nó guarda a atualização de localização numa linha vertical e recupera a localização enviando a consulta numa direção horizontal. Figura 5.10 mostra a consulta propagando na horizontal e encontrando a informação de localização que propagou na vertical. In [Aydin and Shen. 2002], os autores propõem um esquema que combina consumidores (destinos) e produtores (fontes). Os nós são organizados em falsos quorums. Cada nó produtor/consumidor espalha pacotes para formar a forma de uma cruz ('+'). Serviços são combinados na interseção dos nós. Duas formas de cruzes se interceptam em um nó intermediário. Sarkar *et al.* [Sarkar et al. 2006] propuseram métodos de regras duplas onde eles guardam a informação numa curva 1D (e.g., círculo) numa rede de sensores. O consumidor viaja ao longo de outra curva que garanta a interseção com a curva do consumidor.

5.7.2. Método Baseado em Hashing

Em protocolos baseados em hashing, servidores de localização são escolhidos por um função hash no espaço de identificadores dos nós sensores. Esquemas usando hashing podem ser divididos em planos e hierárquicos. Num esquema plano como o GHT [Ratnasamy et al. 2003], a informação de localização é armazenada em uma única localização geográfica. Em um esquema de hashing hierárquico como o GLS [Li et al. 2000], HIGH-GRADE [Yu et al. 2004] e MLS [Flury and Wattenhofer 2006], a área em que os nós residem é recursivamente dividida em hierarquias de grids menores. Em cada nível, um conjunto de nós determinado pela função hash serve como servidor de localização para um dada nó. A figura 5.9 ilustra esse conceito. HIGH-GRADE [Yu et al. 2004] introduz o conceito de "nível de indireção" onde ao invés de armazenar a localização exata, servidores de localização em hierarquia armazenam ponteiros para servidores em níveis menores. MLS [Flury and Wattenhofer 2006] melhora o HIGH-GRADE em termos de número de hops e corretude do protocolo.

5.7.3. Método Baseado em Feromônio

Em [Vieira et al. 2008], os autores desenvolveram um algoritmo inspirado na biologia. Formigas podem explorar vastas áreas sem conhecimento global do terreno. Elas são capazes de encontrar

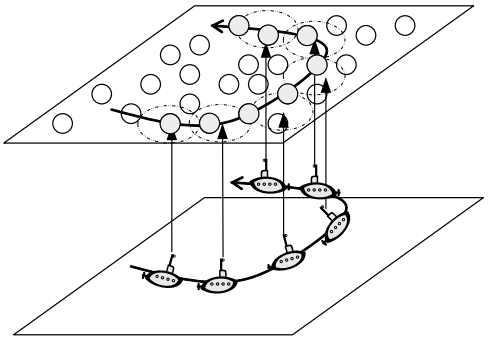


Figura 5.11: Atualização periódica.

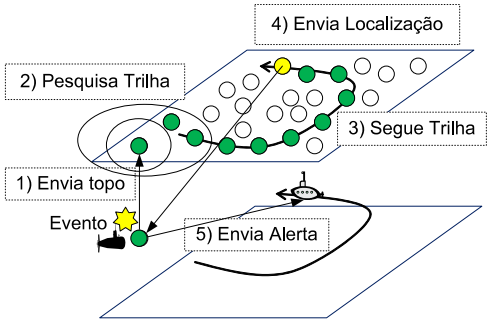


Figura 5.12: Consulta.

comida e a levarem para o formigueiro. Como elas conseguem realizar tais tarefas? Deixando ao longo do caminho rastros de feromônio.

No cenário da aplicação, os veículos aquáticos (que podem ser submarinos) enviam atualizações periódicas de sua localização. A informação de localização é armazenada em um plano 2D. Utiliza-se um roteamento vertical, baseado na coordenada z, aprendida através da pressão, para encaminhar mensagens ao plano onde se localiza o serviço de localização. A atualização periódica cria uma trilha, similar ao feromônio.

O serviço de localização é inspirado na biologia, chamado protocolo de serviço de localização Phero-Trail. A parte superior do casco 2D da topologia do enxame de sensores armazena a informação de localização. Ao contrário de protocolos convencionais 1D que mantêm informação de localização em algumas formas geométricas (linhas ou círculos), os autores propõem o uso da trajetória do nó móvel (à primeira trilha de feromônio das formigas) para apontar para sua localização.

Para atualizar sua localização, o nó móvel (e.g., VAA) encaminha sua posição atual para o ponto da projeção na parte superior do casco 2D. O termo casco é utilizado para diferenciar da superfície. O casco superior é a região superior da topologia formada pelos nós depositados. Ele pode ser a superfície da água se os nós forem depositados a profundidade zero. A sequência de projeções é basicamente equivalente a de uma trilha feromônio. O custo de manutenção da trilha é mínima. O nó móvel (VAA) possui motor próprio e, geralmente, se move muito mais rápido que as correntes carregando os sensores. Assim como a fumaça de rastro de um avião, a trilha de feromônio é lentamente difundida na corrente maritima. O comprimento da trilha é controlado através da configuração de temporizadores.

No caso de uma consulta de localização, o nó sensor envia uma consulta que procura pela trilha. Primeiro, isso é feito com o pacote sendo encaminhado verticalmente para o plano no casco em que se encontra os servidores. Após receber o pacote, o nó no casco realiza uma pesquisa (e.g., caminho aleatório) para encontrar a trilha. Logo depois, o pacote é encaminhado ao longo da trilha para recuperar a localização mais recente do destinatário. Os resultados mostram que, na prática, o método da trilha de feromônio possui os benefícios de métodos hierárquicos sem manter uma hierárquica geográfica.

A figura 5.11 ilustra a atualização da localização. Veículos enviam atualizações periódicas de sua localização. Nós na superfície armazenam essa informação. A projeção da sequência de atualizações forma a trilha. A consulta à localização de um dos veículos aquáticos é feita como mostra a figura 5.12. A consulta é encaminhada verticalmente ao plano onde se encontram os

servidores. A seguir, encontra-se a trilha, que é seguida até encontrar a atualização mais recente. A resposta é enviada ao nó sensor que pode enviar o alerta.

5.8. Modelo de Mobilidade

Os nós sensores podem se mover com as correntes marítimas, permitindo um monitoramento 4D (espaço e tempo) do ambiente. Observações empíricas sugerem que os nós sensores vão se mover a uma velocidade entre 3 e 6 km/h. Nesta seção é descrito um modelo de mobilidade que permite estudar as características de redes de sensores aquáticas, tais como conectividade e cobertura. O modelo de mobilidade é realístico, baseado em medições oceânicas e representado por equações que descrevem o comportamento de fluxos. Além disso, esta seção apresenta os resultados mais recentes em simulação para redes aquáticas que utilizam este modelo.

Para estudar propriedades de sensores interconectados em redes, é crucial que se use um modelo de mobilidade que considere a natureza do fluido do meio em que os nós sensores se movem. Quase todos os modelos existentes na literatura em rede de sensores móveis assumem que cada sensor se move independente dos outros [Bettstetter 2004, Bettstetter et al. 2004]. Tipicamente, o caminho de cada sensor é considerado uma realização de um dado processo estocástico, como passeio aleatório (*random walk*), ou processo de ponto de maneira aleatória (*random way point*). Em um fluído, ao contrário, a mesma velocidade afeta todos os sensores. O caminho deles é determinístico (embora frequentemente caótico), e correlações fortes entre sensores próximos deve ser esperada. Então, para poder simular o movimento dos sensores, é necessário que se modele o movimento dos oceanos em que os sensores estão imersos. Isso pode ser alcançado de várias formas, com diferentes níveis de realismo.

Progresso em entender transporte lagrangeano tem sido feito com uma abordagem puramente cinemática, onde um campo de velocidade razoável é definido de antemão. Para aplicações em redes de sensores aquáticas, se explora o fato que oceanos são um fluído estratificado. Movimentos verticais são, quase em todos os pontos, desprezados quando comparados aos movimentos horizontais [Pedlosky 1996]. Dessa forma, assume-se que os sensores movem-se em uma superfície horizontal e movimentos verticais são desprezados. Modelos desse tipo são bem conhecidos em fluidos dinâmicos, porque eles permitem descrever a cinemática de fluxos quasi-bi-dimensionais de forma simples, enquanto retendo um bom nível de realismo. O livro [Ottino 1989] é uma introdução geral para o leitor interessado, enquanto a monografia recente [Samelson and Wiggins 2006] foca em aplicações geofísicas.

Na oceanografia, a ausência de movimentos verticais é uma característica de projeto de *drifters*, onde os sensores flutuam a uma pequena profundidade fixa dentro de uma bóia perto da superfície [Davis 1985, Sybrandy and Niiler 1991]. No caso de objetos flutuantes (*floats*), as profundidades de operação são usualmente maiores e não existe contato direto com a superfície. O casco dos dispositivos é construído de forma a manter a densidade praticamente constante, para que o objeto flutuante possa ser calibrado e seguir precisamente uma superfície isopícnica [2]. Nesse caso, movimentos verticais dos objetos flutuantes são usualmente limitados a pequenas oscilações ao redor da superfície de referência de densidade e são causados por ondas internas [Rossby et al. 1986].

É claro que, na presença de ventos fortes, ou durante eventos de formação a grandes profundidades, ou na passagem excepcional de ondas internas intensas, a suposição de movimentos verticais desprezíveis deixa de ser válida. Em investigações preliminares, observou-se que esses

[2]Uma superfície de igual densidade. Superfície isopícnicas no oceano são usualmente muito próximas de serem horizontais.

eventos excepcionais poderiam ser desprezados e foi criado um modelo que descreve as condições de circulação da água.

Fluxos bi-dimensionais são descritos por uma função ψ na qual os dois componentes do campo de velocidade sem divergência $\mathbf{u} \equiv (u, v)$ podem ser computados como:

$$u = -\frac{\partial \psi}{\partial y}; \quad v = \frac{\partial \psi}{\partial x}. \tag{11}$$

Por uma convenção de longa data, u é o componente zonal (leste) e v é o meridional (sul). Dessa forma, a trajetória do dispositivo lagrangeano que se move com a corrente é solução do seguinte sistema de equações diferenciais ordinárias hamiltonianas:

$$\dot{x} = -\partial_y \psi(x, y, t), \qquad \dot{y} = \partial_x \psi(x, y, t). \tag{12}$$

Uma função estudada vastamente, que é projetada para pegar as duas principais características de um fluxo oceano típico (correntes e vórtices) foi proposta primeiro por Bower [Bower 1991], que usou o modelo para explicar propriedades de caminhos observados de objetos flutuantes isopícnicos liberados no Atlântico Norte (no *Gulf Stream*). O modelo foi generalizado em [Samelson 1992]. A dinâmica resultante provou ser tão rica e interessante que esses trabalhos deram inicio a um grande número de outros estudos ([Samelson and Wiggins 2006] apresenta uma revisão).

5.8.1. Modelo de Mobilidade de Meandros

Existe a necessidade de modelos de mobilidade para descrever o comportamento de redes de sensores aquáticas. O custo na realização de experimentos é alto. Por exemplo, é necessário adquirir equipamentos, alugar barcos, etc. Em geral, simulações são mais baratas, porém, requerem modelos que descrevam com precisão o comportamento do mundo real. O primeiro modelo de mobilidade para redes aquáticas foi o modelo *Meandering Current Mobility* (MCM) [Caruso et al. 2008]. Este modelo é usado para explicar as propriedades de caminhos observados de flutuantes soltos no *Gulf Stream* a grandes profundidades. A forma adimensional do modelo é:

$$\psi(x, y, t) = -\tanh\left[\frac{y - B(t)\sin(k(x - ct))}{\sqrt{1 + k^2 B^2(t)\cos^2(k(x - ct))}}\right] \tag{13}$$

onde $B(t) = A + \varepsilon\cos(\omega t)$.

Esta função representa uma corrente principal e meandros[3] entre vórtices[4] re-circulantes (ver figura 5.13). A função B, dependente do tempo, modula a amplitude dos meandros. O parâmetro A determina a largura média dos meandros, ε é a amplitude de modulação e ω sua frequência. O parâmetro k define o número de meandros por unidade de comprimento, c é a velocidade do jato principal. Como exemplo significativo, utilizou-se os seguintes valores: A = 1.2, c = 0.12, k = 2π/7.5, ω = 0.4, ε = 0.3.

[3]Meandros são sinuosidades descritas por um curso de água.

[4]Vórtice é um escoamento giratório onde as linhas de corrente apresentam um padrão circular ou espiral. São movimentos espirais ao redor de um centro de rotação.

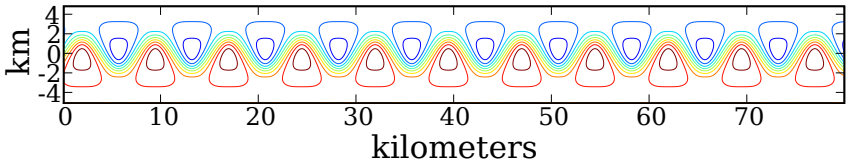


Figura 5.13: Visualização da função dada pela equação (13) no tempo $t = 0$.

A figura 5.14 ilustra a evolução no tempo de uma rede com 100 sensores. Cada ponto na figura representa um nó sensor. Os nós sensores são depositados inicialmente em um quadrado de 4 km de lado no centro do eixo do jato da corrente. A figura mostra a posição dos 100 sensores em vários instantes de tempo, do dia inicial até o terceiro dia. Observa-se que, com o passar do tempo, os nós sensores se movem com as correntes marítimas, que na simulação, seguem o modelo de mobilidade MCM [Caruso et al. 2008]. Ao final de três dias, alguns sensores se encontram a 70 km de distância do ponto inicial, enquanto a maior parte está localizada entre os marcos de 10 e 50 km.

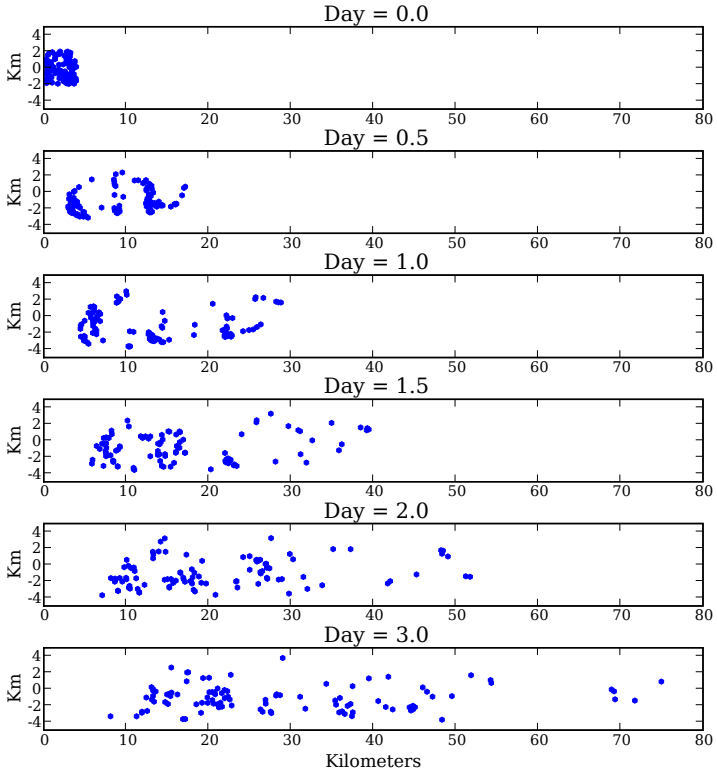


Figura 5.14: Evolução no tempo da posição de 100 sensores. Simulação utilizando o modelo de mobilidade MCM.

5.8.2. Medidas para a Análise do Modelo de Mobilidade

Para estudar o desempenho das redes de sensores, foi introduzido a medida de dispersão absoluta [Provenzale 1999]. A dispersão absoluta ao longo do eixo x-axis é definida como:

$$A^2(t,t_0) = \left\langle |x_i(t) - x_i(t_0)|^2 \right\rangle = \frac{1}{N}\sum_{i=1}^{N} |x_i(t) - x_i(t_0)|^2$$

onde $N = |V|$ é o número de sensores na rede, $\langle \ldots \rangle$ indica a média sobre os nós sensores, $x_i(t)$ é a coordenada x do $i-$th sensor no tempo t, t_0 é o tempo de deposição. Em [Caruso et al. 2008], há resultados da média do A^2 em diferentes realizações do mesmo processo de deposição. A média do A^2 provê uma medida da rede para a dispersão dos nós em função do tempo. A forma como a dispersão absoluta escala com o tempo caracteriza a natureza física do processo de transporte: se $A^2 \propto t$ então o processo é difusivo; se $A^2 \propto t^2$ então temos um processo de transporte balístico [Provenzale 1999].

Note que, em uma rede aquática móvel, o efeito combinado de uma potência limitada de transmissão, mobilidade sobre uma grande área, e limitado alcance de comunicação, implica que a comunicação irá precisar de múltiplos saltos. Mais ainda, com grande probabilidade o grafo de comunicação G será particionado em vários componentes conectados. Para superar esse efeito, técnicas de roteamento de redes tolerantes ao atraso (DTN, em inglês *delay tolerant networking*) podem ser usadas [Fall 2003]. Por essa razão, a análise de dispersão dos nós pertencentes ao maior componente conectado (*LCC*, do inglês *largest connected component*) é de interesse particular.

Denotando como $LCC(t)$ o conjunto de nós sensores no maior componente conectado no tempo t, as funções de caixa delimitadora foram definidas como [Caruso et al. 2008] $x^{lcc}_{min}(t) = \min_{i\in LCC(t)} x_i(t)$ e $x^{lcc}_{max}(t) = \max_{i\in LCC(t)} x_i(t)$. A caixa delimitadora $LCC(t)$ foi comparada com a caixa delimitadora de toda a rede , i.e. $x^{G}_{min}(t) = \min_{i\in V(t)} x_i(t)$ e $x^{G}_{max}(t) = \max_{i\in V(t)} x_i(t)$.

5.8.3. Conectividade

A conectividade entre os nós sensores é afetada pelo modelo de mobilidade. É de interesse científico estudar a dispersão dos nós depois de uma deposição, e como a dispersão e mobilidade afetam a densidade e conectividade da rede. O *LCC* é utilizado como uma medida de conectividade.

A figura 5.15 mostra a evolução no tempo (em horas) de diferentes métricas de conectividade da rede. No eixo y se encontra o valor da coordenada x (em km). Logo após a deposição, a rede está conectada, e claramente $x^{G}_{min}(t) = x^{lcc}_{min}(t)$ e $x^{G}_{max}(t) = x^{lcc}_{max}(t)$. A função $x^{G}_{max}(t)$ segue a mesma trajetória que a função $x = v_m t$, que significa que alguns sensores estão no meio da corrente de meandros e se movem com velocidade v_m. Depois de um tempo, dependendo da densidade da rede e do alcance de comunicação, a rede se torna desconecta (o que explica a queda no $x^{lcc}_{max}(t)$). Uma fração de sensores continua a mover-se com velocidade v_m, determinando o valor máximo da caixa delimitadora para toda a rede ($x^{G}_{max}(t) \sim v_m t$), enquanto outra fração permanece em vórtices determinando o mínimo da caixa delimitadora para toda a rede ($x^{G}_{min}(t)$). Nós pertencentes ao conjunto LCC se movem com uma velocidade média menor que v_m; essa redução na velocidade é explicada pelo fato que muitos sensores passam uma boa parte do tempo circulando dentro dos vórtices

5.8.4. Cobertura

A área de sensoriamento é a área que um nó pode sensoriar o ambiente e detectar eventos, e pode ser modelada por um disco de raio R_s centrado na posição do nó sensor. Um ponto do domínio é *coberto* por um sensor se sua localização for dentro da área de sensoriamento de algum nó.

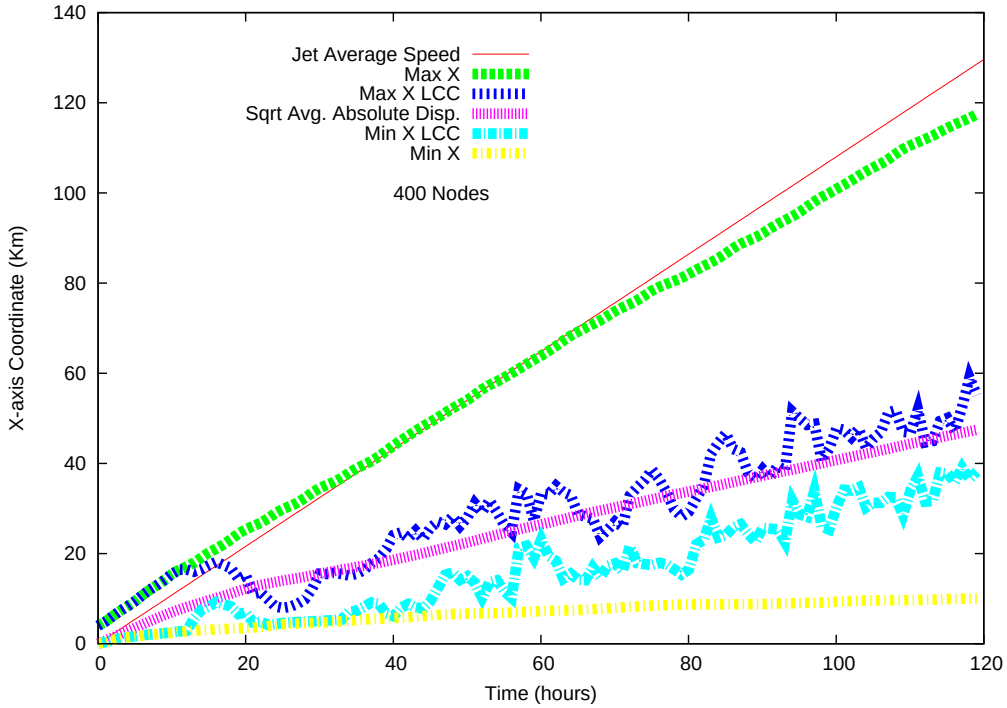


Figura 5.15: Caixa delimitadora de toda a rede, do $LCC(t)$, raiz quadrada da dispersão absoluta $A(t,t_0)$ e $x = v_m t$. Número de sensores $N = 400$.

Para cada distribuição estática dos nós, o domínio pode ser particionado em duas áreas: a região coberta, que é o conjunto de pontos coberto por pelo menos um sensor, e a região não-coberta definida como o complemento da região coberta. Duas medidas são utilizadas [Liu et al. 2005] para cobertura estática e móvel:

Definição 5.8.1 (Área Coberta) *A área coberta de uma rede de sensores no tempo t, $f_a(t)$ é a fração de área geográfica coberta por pelo menos um ou mais nó sensor no tempo t.*

Definição 5.8.2 (Área Coberta sobre um intervalo de tempo) *A área coberta de uma rede móvel de sensores durante o intervalo de tempo $[0,t)$, $f_m(t)$ é a fração de área geográfica coberta por pelo menos um sensor em algum ponto de tempo dentro de $[0,t)$.*

A área coberta é importante em aplicações que requerem cobertura simultânea do domínio geográfico. A cobertura $f_m(t)$ é mais apropriada para aplicações que não requerem cobertura simultânea de todos os pontos num dado instante de tempo específico, mas prefere cobrir a rede em um intervalo de tempo.

Resultados de simulação mostram a área de cobertura definida previamente. Nós são uniformemente distribuídos na área de deposição e depois eles dispersam pelo domínio seguindo as correntes do oceano. Na simulação, foi utilizado o modelo de disco para a área de sensoriamento, com um raio de sensoriamento igual a $R_s = 0.5$km para cada nó sensor.

A figura 5.16 mostra o impacto do movimento dos nós na área de cobertura, para redes com um número crescente de nós. A dispersão dos nós da área de deposição para todo o domínio aumenta a área coberta da rede.

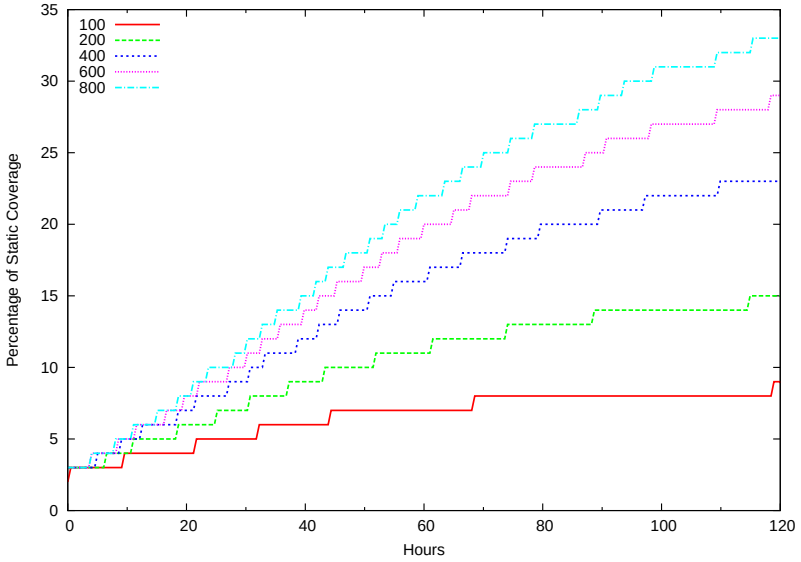


Figura 5.16: Área coberta pela rede. O domínio é D=[0,80]x[-4,4].

Em qualquer caso, a área de cobertura máxima é apenas uma fração de todo o domínio. Na figura 5.17 observa-se a área coberta pelo tempo. Dado um tempo t, o gráfico representa a fração da área do domínio que tem sido coberta por pelo menos um instante de tempo no intervalo $[0,t)$. Do ponto de vista gráfico é possível concluir que mobilidade aumenta a cobertura "dinâmica". Como esperado, ela também aumentar com o aumento de densidade.

5.8.5. Deposição

É de interesse científico descobrir a melhor forma de depositar os sensores, formando a rede. Uma modelagem inicial da implantação de sensores na rede é como um processo finito discreto aleatório numa região, por exemplo, $4\times\ 4\,\mathrm{km}^2$. Considere um número fixo de sensores N. Inicialmente, mostramos o resultado de um estudo com dois processos simples:

1. Uma única fase de deposição de N nós.
2. Processo de duas fases.

O estudo do processo de duas fases mostrou que o tempo de espera entre duas deposições é relevante. Deseja-se maximizar a cobertura de rede. Dessa forma, quanto maior o intervalo de tempo entre duas implantações, maior é a cobertura de rede porque os nós sensores se movem, e com o passar do tempo, cobrem uma região maior. Também se deseja manter a maior parte dos nós sensores da rede conectados. Nesse caso, quanto menor o tempo entre duas implantações, maior é a chance de conectividade visto que os nós sensores da primeira implantação se mantêm próximos aos da segunda deposição. Esses dois fatos geram um compromisso entre cobertura e conectividade na deposição dos nós. Tópicos de pesquisa incluem novos modelos de deposição e estudos com diferentes fases de implantações.

A primeira queda no $x_{max}^{lcc}(t)$ corresponde a primeira vez que a rede fica desconectada. Este tempo é uma função do número de nós (densidade da rede), alcance de comunicação, e da velocidade do jato principal. Foi estudado, em particular, o primeiro instante de tempo que o número de sensores no *LCC* se torna menor que $90\%N$ (T_{conn}^{90}), em outras palavras, uma medida do tempo de vida da rede. Na figura 5.18 está o gráfico de T_{conn}^{90} para diferentes valores de N. Para

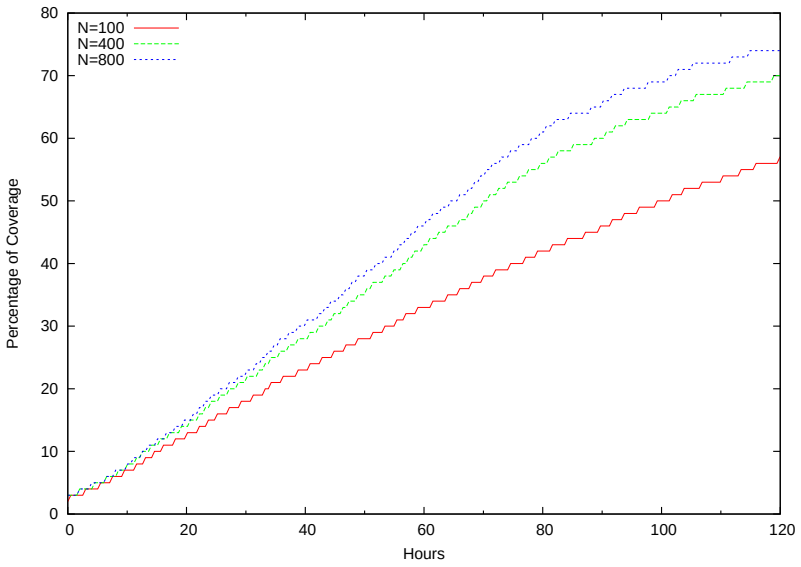


Figura 5.17: Área coberta por um intervalo de tempo.

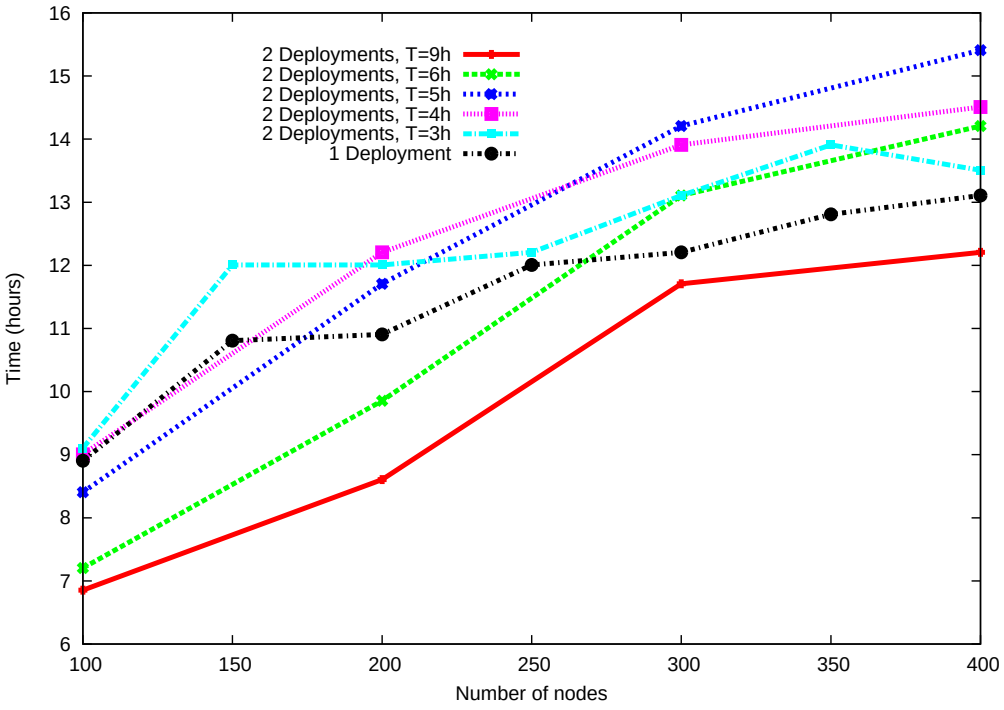


Figura 5.18: Conectividade: primeira vez que o tamanho do *LCC* cai para menos de 90%N. Deposições com 1 ou 2 etapas, e diferentes valores de ΔT.

um valor fixo de N, foram considerados deposições em uma etapa e duas etapas. No caso de duas etapas, $N/2$ nós foram depositados no $T = 0$ e o resto $N/2$ no tempo $T = \Delta T$, variando ΔT.

Com uma única deposição, o valor de T_{conn}^{90} está no intervalo $[9-13]$h e está claro que é uma função crescente em N. Com a deposição em duas etapas, se $\Delta T = 3$h ou 4h, o valor de T_{conn}^{90} aumenta e, em alguns casos, para redes grandes ($N = 800$), é 20% maior que valor correspondente para uma única deposição.

Se o valor de ΔT for muito alto ($\Delta T = 9$h), o valor de T_{conn}^{90} é sempre menor que o valor correspondente para uma deposição, porque os nós na segunda etapa não conseguem alcançar os nós da primeira deposição.

5.9. Aplicações

Esta seção apresenta algumas aplicações para redes de sensores aquáticas. Várias áreas que podem se beneficiar do desenvolvimento das redes de sensores aquáticas, como oceanografia, biologia marinha, arqueologia, segurança, estudo do aquecimento e da acidez dos oceanos, estudo da interação entre oceanos e atmosferas, compreensão da formação de tsunamis, predição sísmica, detecção de poluentes, detecção de submarinos e mergulhadores, monitoração de derramamento de substâncias químicas ou óleo, etc. Nesta seção são discutidos mais detalhadamente aplicações em sismologia, segurança, poluição e biologia marinha.

5.9.1. Sismologia

Sismologia é o estudo dos terremotos e ondas elásticas na Terra. É um ramo da geo-física dedicado a estudar fontes sísmicas como vulcões, placas tectônicas, terremotos, tectonic and earthquakes como também propriedades físicas da Terra. Se se deseja olhar dentro da Terra, medições de ondas sísmicas são a forma mais precisa e, portanto, o método preferido. Cientistas usam fontes artificiais, como explosões, e estudam reflexão e refração das ondas em certas áreas para determinar a estrutura geológica, como por exemplo na busca por hidrocarbonetos. Atividades em sismologia podem ser resumidas em:

- Análises e monitoração de terremotos.
- Estudo da estrutura da Terra e suas propriedades físicas.
- Uso de fontes artificiais para obter informações sobre bases sedimentares na procura por minerais e hidrocarbonetos, e medidas da grossura da camada de gelo em mares e geleiras.
- Inquéritos raso por motivos de hidrologia e construção.
- Sismologia teóricas e processamento de dados.

No século XX, investigação militar sobre ondas de superfície e o advento da tomografia permitiram o pioneirismo da exploração sismológica. Os cientistas mediram a velocidade sísmica, e estudaram o tempo de viagem de ondas em pesquisas com fontes artificiais para criar mapas de estruturas geológicas. A técnica comum é o método de reflexão onde, reflexões verticais são medidas com uma rede distribuída de sismógrafos. Os sismógrafos registram o movimento do solo, com um efeito de ondas sísmicas, convertem o movimento em tensão. Em pesquisa no fundo do mar, hidrofones são mais comumente usados, eles são colocados em cabos longos atrás do navio de pesquisa e convertem a pressão em tensão. Outra possibilidade em levantamentos sísmicos marinhos é arrastar cordas ao longo do fundo do mar para registrar ondas de pressão. Figura 5.19 mostra uma exploração sísmica convencional. O barco usa uma arma de ar para causar

ondas de choques que são refletidas pelas camadas geológicas. Também são utilizados hidrofones para registrar as ondas de pressão refletidas.

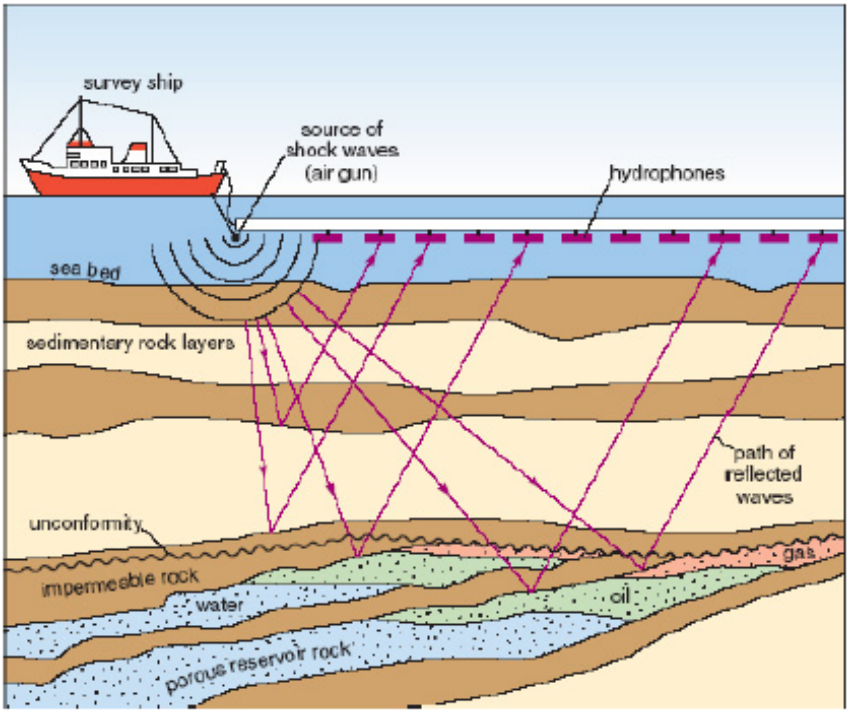


Figura 5.19: Exploração sísmica utilizando reflexão de ondas e hidrofones.

O objetivo básico de um sismógrafo é gravar o movimento do solo, em relação a um ponto estacionário, acima do solo. Em instrumentos iniciais, o movimento do solo era gravado por um pêndulo traçado na areia ou em papel. Com o avanço da tecnologia, o pêndulo amortecido horizontal provou ser mais exato. Este inclui um peso montado no ponto de um triângulo bem articulado em sua borda vertical. Quando o chão está se movendo, o peso permanece imóvel e a dobradiça balança. A oscilação do pêndulo pode ser ajustada para se obter a frequência desejada. No estado da arte, sismógrafos utilizam sensores eletrônicos, amplificadores e filtros para obter uma alta resolução e detectarem eventos em um espectro de banda larga.

Após o terremoto no Oceano Índico em 2004, e o tsunami devastador seguinte a ele, a pesquisa sobre terremotos submarinos e detecção de tsunamis ganhou mais atenção. A monitoração das placas tectônicas profundas do mar, por muitos anos, tem sido uma área de foco para os cientistas sísmicos. Mais de 80% de todos os terremotos ao redor do Japão ocorrem no fundo do mar por subducção de placas, que podem ser devastadoras para o continente, no caso de um incidente de tsunami. No Japão, há atualmente algumas rede fixas de monitoração em operação no fundo do mar. Cada rede é composta por vários sensores de pressão, para detecção de tsunamis, e sismógrafos no fundo do oceano, ligados via cabo a uma central de monitoração. O objetivo dessas redes é, naturalmente, relatar qualquer atividade sísmica precursora ou deslizamento entre as placas, dando boas indicações sobre terremotos próximos. A figura 5.20 mostra a configuração de um sismógrafo no fundo do mar e sua foto real.

Redes de sensores aquáticas, com algoritmos de boa localização, podem ser um bom método para reduzir custos e simplificar a implantação de sistemas de monitoração sísmicos, por sua estratégia ad hoc.

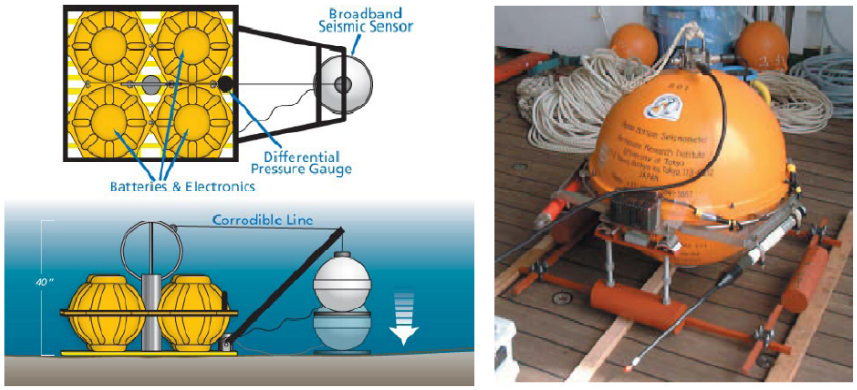


Figura 5.20: Sistema de sismógrafos no fundo do mar.

5.9.2. Segurança

A tecnologia está caminhando para exploração dos recursos naturais e outros tipos de infra-estruturas marítimas, como plantas de geração de energia via ondas e culturas de peixe. Tudo isso sem a proteção de submarinos. Esses tipos de instalações são muito complexos e consistem de diferentes construções espalhadas por uma grande área. Atualmente, esses equipamentos são altamente vulneráveis a ataques terroristas e não convencionais de guerra. Outro alvo em potencial são portos cuidando de centenas de embarcações marítimas a cada dia. Um ataque contra qualquer infra-estrutura marítima pode ter efeito devastador na economia e no meio-ambiente. Isso faz com que a área de segurança na orla deixe de ser uma preocupação exclusivamente militar.

Instalações militares estão, em geral, sob a segurança e bem protegidas, tanto debaixo da água como na superfície. A segurança de portos ganhou muita atenção nos últimos anos e o uso de veículos submarinos autônomos para uso comercial permitiu o patrulhamento subaquática eficaz de grandes zonas portuárias.

Ameaças, em potencial, incluem mergulhadores, submarinos não tripulados, veículos, barcos de velocidade e VAA. Todas as possíveis ameaças têm características diferentes e possíveis abordagens e trajetórias que requerem um sistema avançado de detectar todos os tipos de ameaças. Um dos maiores desafios é o de ser capaz de identificar ameaças escondidas dentro da atividade diária, potencialmente disfarçada como um navio de pequeno porte.

Um sistema de segurança para proteção das instalações marítimas precisa servir várias funções para alcançar o mesmo nível de segurança de instalações importantes em terra [Lougheed and Clifton 1988]. A tabela 5.2 dá uma visão geral e exemplos.

Na luta contra ameaças, o sonar subaquático é a ferramenta preferida. Sistemas de sonar podem ser ativos ou passivos. Sistemas ativos usam um ou mais transmissores e receptores, enquanto um sistema sonar passivo utiliza apenas receptores. Os transmissores de um sistema de sonar ativo emite um pulso curto e analisa o eco recebido para identificar todos os navios, submarinos ou obstáculos. Um sistema de sonar passivo, por outro lado só ouve o ruído emitido por alvos em potencial, e identifica o alvo baseado em assinaturas sonoras distintas. A maioria dos sonares são montadas sobre o casco de um navio ou carregados atrás dele.

Os sistemas de vigilância para a proteção de instalações em alto mar são geralmente baseados em uma abordagem integrada [Asada et al. 2007]. Isto envolve o uso de vários sistemas de sensoriamento remoto e plataformas de sensores. Para a vigilância, radar acima da superfície e barcos de patrulha são comuns, e vários sistemas de sonar são empregadas sob a superfície. Mui-

Tabela 5.2: Funções típicas de um sistema de segurança marítima.

Função	Terra	Mar
Prevenção	Trancas, cercas, controle de acesso.	Barreiras de controle de acesso, iluminação.
Detecção	Sensores de detecção, vídeo de detecção de movimentos.	Sonar, radar, vídeo de detecção de movimentos.
Avaliação	Televisão de circuito fechado (CCTV)	CCTV, veículos subaquáticos
Impedimento de acesso	Cercas, barreiras	barreiras flutuantes.

tas vezes os sistemas de vigilância são combinados com barreiras flutuantes e pontos de controle para fins de prevenção. A figura 5.21 ilustra um sistema de vigilância: veículos aquáticos utilizam sonar para detectar mergulhadores acima deles e múltiplos sonares com eco para detectar cascos de navios.

Figura 5.21: Sistema de vigilância com sonares simples e duplos.

Um dilema na aplicação de sistemas de vigilância subaquática diz respeito ao custo e à cobertura [Smookler et al. 2005]. Um sistema básico para detecção de mergulhador, que é o mais adequado para proteção contra ataques assimétricos, utilizam sonares com bandas de frequência mais alta. A alta frequência sofre maior perda de absorção [Hovem 2008], limitando assim o alcance de proteção. Isto exige uma grande quantidade de instrumentos distribuídos para conseguir uma boa cobertura, o que acaba por aumentar o custo. Uma abordagem proposta é combinar os sistemas de detecção de mergulhador próximo às instalações e veículos ou sonares rebocados para patrulhamento aleatório mais longe da instalação [Lovik et al. 2007]. Mesmo esta solução, que não oferece cobertura completa, pode aumentar a probabilidade de detecção em uma área maior.

Uma abordagem utilizando redes ad hoc para uma boa cobertura na vigilância marítima de áreas de alto risco é apresentado em [Benmohamed et al. 2006]. A rede de sensores é composta por um número de nós receptores capazes de analisar sinais acústicos. São utilizados alguns nós com transmissores acústicos e um gateway com enlace de satélite. Os nós têm um receptor acústico

e uma unidade de processamento para analisar sinais recebidos. Quando a detecção é feita, um relatório é enviado para um centro de controle em terra para analisar o alarme.

5.9.3. Poluição

O oceano é a estação final para grandes quantidades de lixo. Na maioria das vezes, a poluição marinha é proveniente de fontes terrestres, incluindo lixo doméstico, esgoto, escoamentos agrícolas e diferentes substâncias químicas tóxicas. Toxinas podem originar, por exemplo, de vento levando pesticidas ou rios poluídos. No oceano, elas se dissolvem em muitas partículas pequenas e são absorvidos pelos plânctons. Uma vez no ecossistema, a poluição se concentra dentro da cadeia alimentar e contamina todas as partes da vida marinha. Muitas partículas químicas carregadas por rios podem combinar-se quimicamente de forma a tornar estuários anóxicos, exterminando toda vida aquática. Como os recursos marinhos são usados para alimentação, metais pesados e substâncias químicas perigosas podem, eventualmente, acabar em seres humanos ou se espalharem para os animais terrestres. Além da disseminação de produtos tóxicos e de mortalidade, a poluição também pode causar mutações ou alteração na bioquímica, modificando o tecido com capacidade de reprodução [Owen et al. 2005].

O aumento do tráfego de navios e petróleo, em alto mar, e a extração de gás são uma grande preocupação. Boa parte da poluição no oceano é originada de descarga de navios e acidentes em navios e plataformas. A quantidade de óleo derramada, anualmente, em todo o mundo, é estimada em mais de 4.5 milhões de toneladas, o equivalente a dois acidentes com super petroleiros a cada semana. Mas descargas e derrames de petróleo estão diminuindo. Desde a década de 1980, elas foram reduzidas em 63% e continuam a diminuir. Isso se deve, principalmente, ao advento de petroleiros de casco duplo e iniciativa política [Khan and Jenkins 2008].

A preocupação do nível de CO_2 na atmosfera preocupa muitos cientistas do clima. Dado que a maior quantidade de CO_2 acaba no oceano, a medição de seu nível na água dos oceanos, pode dar boas indicações sobre a parcela do aumento de descarga terminando na atmosfera. Como o aumento do teor de CO_2 na atmosfera é a metade do que previsto, a pergunta que fica é se os oceanos absorvem mais do que era inicialmente esperado [Rubin and Ping Wu 2000]. Para ser capaz de fazer previsões futuras do nível atmosférico de CO_2, a transferência líquida de CO_2 para a superfície das águas tem que ser medida. Isso pode ser feito através da medição da pressão parcial de CO_2 (pCO_2) e calculada a diferença da pressão na camada da atmosfera, o que é bastante constante. Esta diferença de pCO_2 decide o coeficiente de transferência do gás e permite calcular a transferência líquida de CO_2 para superfície da água. Medições de pCO_2 na superfície da água é feito atualmente por missões de investigação a nível mundial, mas esta abordagem não tem a capacidade de fazer o acompanhamento detalhado no tempo e no espaço que é necessário. Para confirmar a necessária amostragem espacial e temporal, sensores de fibras ópticas capazes de monitoração de longo prazo têm sido desenvolvidos [Rubin and Ping Wu 2000].

A necessidade de amostragem adaptativa de poluentes, em determinadas áreas marítimas, é uma realidade. A falta de métodos mais eficazes para detecção, monitoração, análise e previsão da poluição marinha é de grande importância. A metodologia atual de amostragem ativa e passiva pode não ser aplicável a acompanhar a propagação futura de poluentes marinhos. Para avaliar a ameaça iminente da poluição marinha, métodos de amostragem adaptativa devem ser desenvolvidos.

Para o acompanhamento de limites costeiros e estuários, redes aquáticas podem ser uma boa solução. Redes aquáticas podem ser equipadas com sensores para detecção de turbidez[5], nível

[5]A turbidez é uma característica física da água, decorrente da presença de substâncias em suspensão e é

de oxigênio, temperatura e pH. Em caso de uma emergência, uma rede ad hoc aquática pode ser depositada rapidamente e ser usada para monitoramento de derrames de petróleo. Monitoramento de descargas das instalações em alto mar também são adequadas. Redes aquáticas podem prover sensoriamento quase em tempo real, podendo ser útil também em áreas de alto risco.

5.9.4. Biologia marinha

Biologia marinha é o estudo de organismos vivos no mar ou em outros ambientes marinhos. Além de estudar os diferentes organismos que compõem o elenco no mar, biólogos estudam o efeito de diferentes substâncias sobre a vida marinha. Consequentemente, é muito relacionada com a oceanografia e o estudo da salinidade, da temperatura e de mudanças dinâmicas do mar.

Como a oceanografia, a biologia marinha é uma ciência relativamente nova e originou-se do estudo das criaturas terrestres. Uma motivação importante para o estudo da vida no mar é a hipótese que a vida começou no mar. Biologia marinha cobre desde o estudo de organismos unicelulares e fotossíntese de bactérias em águas profundas a migração dos grandes mamíferos marinhos.

Os cientistas acreditam que a vida no mar nos diz muito sobre a evolução da Terra e dão boas indicações sobre as mudanças ocorridas no passado, e as mudanças que ocorrem agora. A relação entre a vida no mar e mudanças importantes acontecendo na atmosfera ainda não está totalmente compreendida. Grande parte do oceanos ainda permanecem inexplorados e podem conter informações novas sobre a evolução e os recursos futuros. Os recursos marinhos são de grande importância para a qualidade de vida e para a vida na Terra.

Uma parte vital da biologia é o estudo dos habitats e ecossistemas. A mudança física ou química, em uma área específica, pode ter um impacto importante sobre a quantidade de vida na região. Acompanhamento de um habitat pode fornecer bons dados sobre a abundância das espécies e condições para a vida ou reprodução. Habitats a serem monitorados incluem: camadas de águas superficiais sofrendo mudanças rápidas, os recifes com um ambiente rico, com centenas de espécies de águas profundas e trincheiras nas profundidades de vários milhares de metros e sem luz solar [Ingmanson and J.Wallace 1995].

Redes aquáticas podem simplificar e limitar os custos de monitoramento de habitats marinhos. O fato de habitats oceânicos serem limitados em tamanho os torna uma plataforma adequada para o uso de redes de sensores aquáticas. Um recife pequeno, em uma lagoa ou outro habitat de interesse não precisa ter uma rede com mais do que cem nós, tornando a rede mais fácil de ser gerenciada e controlada. O tempo de vida das redes é dependente da amostragem temporal e, consequentemente, da energia relacionada com o armazenamento, processamento e comunicação dos dados.

Atualmente, os sistemas de monitoração são fixos, tendo pequena área de cobertura, pequena proteção (alguns pontos de observação que podem ser facilmente atacados). O uso de redes de sensores pode aumentar a área de cobertura, aumentar a segurança e reduzir custos.

5.10. Conclusões

Rede de sensores aquática é uma área de pesquisa importante. Há uma perspectiva de seu crescimento e utilização em diversas áreas nos próximos anos, como: oceanografia, biologia marinha,

---

medida através da redução de transparência na água. Os equipamentos mais utilizados para medir a turbidez são os nefelômetros que medem, numa célula fotoelétrica, a quantidade de luz dispersa através da amostra de água, a 90^o da luz incidente.

estudos da interação entre oceanos e atmosfera, estudos do clima, aquecimento global, arqueologia no fundo do mar, predições sísmicas, na área de saúde para detecção de poluentes e substâncias contaminantes, controlando a qualidade da água, e áreas importantes da economia como exploração e monitoração de campos de gás, óleo e petróleo. Neste texto, foram apresentados o estado da arte, e particularidades da área de pesquisa em questão.

A principal diferença das redes de sensores aquáticas para as terrestres é o meio de comunicação. Esse aspecto acaba por influenciar o desenvolvimento das aplicações e todas as camadas de protocolos. O texto descreveu a camada física, onde foi detalhada a propagação de sinal e formulado a estimativa da entrega de pacotes no meio aquático. Na camada de enlace, foram apresentados os protocolos mais recentes nesse tema de pesquisa. A camada de roteamento discutiu protocolos pró-ativos, reativos e geográficos. Também foram apresentados métodos recentes de localização: com ajuda de veículos aquáticos, com sinalizadores que ascendem e descendem e, com uso de laser acústico. Resultados recentes em modelos de mobilidade descreveram o modelo MCM. Finalmente, quatro aplicações foram detalhadas: sismologia, segurança, poluição e biologia marinha.

Referências

[OSU] Nsf: Observatório oceânico `http://www.nsf.gov/news/news_summ.jsp?cntn_id=115444\&org=NSF\&from=news`.

[OLS] Optimized link state routing protocol, ietf rfc 3626, oct. 2003.

[Akyildiz et al. 2005a] Akyildiz, I. F., Pompili, D., and Melodia, T. (2005a). Underwater acoustic sensor networks: research challenges. *Ad Hoc Networks*, 2(3):257–279.

[Akyildiz et al. 2005b] Akyildiz, I. F., Pompili, D., and Melodia, T. (2005b). Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks (Elsevier)*, 3(3):257–279.

[Asada et al. 2007] Asada, A., Maeda, F., Kuramoto, K., Kawashima, Y., Nanri, M., , and Hantani, K. (2007). Advanced surveillance technology for underwater security sonar systems. In *Oceans*, pages 01–05.

[Aydin and Shen. 2002] Aydin, I. and Shen., C. (2002). Facilitating Match-Making Service in Ad hoc and Sensor Networks Using Pseudo Quorum. In *ICCCN*.

[Benmohamed et al. 2006] Benmohamed, L., Chimento, P., Doshi, B., Henrick, B., and Wang, I.-J. (2006). Sensor network design for underwater surveillance. pages 1 –7.

[Bertsekas and Gallager] Bertsekas, D. and Gallager, R. Data networks.

[Bettstetter 2004] Bettstetter, C. (2004). *Mobility Modeling, Connectivity, and Adaptive Clustering in Ad Hoc Networks*. Utz Verlag.

[Bettstetter et al. 2004] Bettstetter, C., Hartenstein, H., and Perez-Costa, X. (2004). Stochastic properties of the random waypoint mobility model. *Wireless Networks*, 10(5):555–567.

[Bharghavan et al. 1994] Bharghavan, V., Demers, A., Shenker, S., and Zhang, L. (1994). Macaw: a media access protocol for wireless lan's. In *SIGCOMM '94: Proceedings of the conference on Communications architectures, protocols and applications*, pages 212–225, New York, NY, USA. ACM.

[Blackmon and Antonelli 2006] Blackmon, F. and Antonelli, L. (2006). Remote, aerial, translayer, linear and non-linear downlink underwater acoustic communication. In *OCEANS 2006*, pages 1–7.

[Bose et al. 1999] Bose, P., Morin, P., Stojmenović, I., and Urrutia, J. (1999). Routing with guaranteed delivery in ad hoc wireless networks. In *DIALM '99: Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 48–55, New York, NY, USA. ACM.

[Bower 1991] Bower, A. S. (1991). A simple kinematic mechanism for mixing fluid parcels across a meandering jet. *J. Phys. Ocean.*, 21(1):173–180.

[Brekhovskikh and Lysanov 2003] Brekhovskikh, L. M. and Lysanov, Y. (2003). *Fundamentals of Ocean Acoustics*.

[Carbonelli and Mitra 2006] Carbonelli, C. and Mitra, U. (2006). Cooperative multihop communication for underwater acoustic networks. In *WUWNet '06: Proceedings of the 1st ACM international workshop on Underwater networks*, pages 97–100, New York, NY, USA. ACM.

[Caruso et al. 2008] Caruso, A., Paparella, F., Vieira, L. F. M., Erol, M., and Gerla, M. (2008). The meandering current mobility model and its impact on underwater mobile sensor networks. In *INFOCOM*, pages 221–225.

[Clausen and Jacquet 2003] Clausen, T. and Jacquet, P. (2003). Optimized link state routing protocol (olsr).

[Davis 1985] Davis, R. (1985). Drifter observations of coastal surface currents during CODE: the method and descriptive view. *J. Geophys. Res.*, (90):4741–4755.

[Erol et al. 2007a] Erol, M., Vieira, L. F. M., and Gerla, M. (2007a). AUV-Aided Localization for Underwater Sensor Networks. In *WASA'07*, Chicago, IL.

[Erol et al. 2007b] Erol, M., Vieira, L. F. M., and Gerla, M. (2007b). Localization with Dive'N'Rise (DNR) Beacons for Underwater Acoustic Sensor Networks. In *WUWNet'07*, Montreal, Quebec, Canada.

[Fall 2003] Fall, K. (2003). A delay-tolerant network architecture for challenged internets. In *SIGCOMM '03*, pages 27–34, Karlsruhe, Germany.

[Flikkema] Flikkema, P. Spread-spectrum techniques for wireless communication.

[Flury and Wattenhofer 2006] Flury, R. and Wattenhofer, R. (2006). Mls: an efficient location service for mobile ad hoc networks. In *MobiHoc '06: Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, pages 226–237, Florence, Italy. ACM.

[Flury and Wattenhofer 2008] Flury, R. and Wattenhofer, R. (2008). Randomized 3D Geographic Routing. In *INFOCOM'08*, Phoenix, AZ.

[Freitag et al. 2005] Freitag, L., Grund, M., Singh, S., Partan, J., Koski, P., and Ball, K. (2005). The whoi micro-modem: an acoustic communications and navigation system for multiple platforms. pages 1086–1092 Vol. 2.

[Friedman and Kliot 2006] Friedman, R. and Kliot, G. (2006). Location services in wireless ad hoc and hybrid networks: A survey. In *Technical Report CS-2006-10*.

[Fullmer and Garcia-Luna-Aceves 1995] Fullmer, C. L. and Garcia-Luna-Aceves, J. J. (1995). Floor acquisition multiple access (fama) for packet-radio networks. In *ACM SIGCOMM 95*.

[Füßler et al. 2003] Füßler, H., Käsemann, M., Mauve, M., Hartenstein, H., and Widmer, J. (2003). Contention-based Forwarding for Mobile Ad-hoc Networks. *Elsevier Ad Hoc Networks*, 1(4):351–369.

[Haas and Liang 1999] Haas, Z. J. and Liang, B. (1999). Ad Hoc Mobility Management with Uniform Quorum Systems. *IEEE Transaction on Networking*, 7(2):228–240.

[Hovem 2008] Hovem, J. M. (2008). Marine acoustics part i. In *Norwegian University of Science and Technology*.

[Ingmanson and J.Wallace 1995] Ingmanson, D. E. and J.Wallace, W. (1995). *Oceanography : an introduction*. Wadsworth, San Diego.

[Jalving 1999] Jalving, B. (1999). Depth accuracy in seabed mapping with underwater vehicles.

[Johnson et al.] Johnson, D., Maltz, D., and Broch, J. *DSR The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks*.

[Jones et al. 2006] Jones, T., Ting, A., Peano, J., Sprangle, P., and DiComo, G. (2006). Remote underwater ultrashort pulse laser acoustic source. In *CLEO/QELS 2006.*, Long Beach, CA.

[Kalosha et al. 2008] Kalosha, H., Nayak, A., Ruhrup, S., and Stojmenovi, I. (2008). Select-and-Protest-Based Beaconless Georouting with Guaranteed Delivery in Wireless Sensor Networks. In *INFOCOM'08*, Phoenix, AZ.

[Karn 1990] Karn, P. (1990). Maca : A new channel access protocol for packet radio. In *ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, pages 134–140.

[Karp and Kung 2000] Karp, B. and Kung, H. T. (2000). Gpsr: greedy perimeter stateless routing for wireless networks. In *MobiCom '00*, pages 243–254, Boston, Massachusetts, USA.

[Khan and Jenkins 2008] Khan, A. and Jenkins, L. (2008). Undersea wireless sensor network for ocean pollution prevention. pages 2 –8.

[Kilfoyle and Baggeroer 2000] Kilfoyle, D. B. and Baggeroer, A. B. (2000). The state of the art in underwater acoustic telemetry. *IEEE J. of Oceanic Enginnering*, 25(1):4–27.

[Kleinrock and Tobagi 1975] Kleinrock, L. and Tobagi, F. A. (1975). Packet switching in radio channels: Part i - carrier sense multiple access modes and their throughput- delay characteristics. In *Communications, IEEE Transactions on*, volume 23, pages 1400–1416.

[Kong et al. 2005] Kong, J., Cui, J., Wu, D., and Gerla, M. (2005). Building underwater ad-hoc networks and sensor networks for large scale real-time aquatic applications. In *IEEE MILCOM*, Atlantic City, NJ, USA.

[Kredo et al. 2009] Kredo, K., Djukic, P., and Mohapatra, P. (2009). Stump: Exploiting position diversity in the staggered tdma underwater mac protocol.

[Langendoen and Reijers 2003] Langendoen, K. and Reijers, N. (2003). Distributed localization in wireless sensor networks: a quantitative comparison. *Comput. Networks*, 43(4):499–518.

[Lee et al. 2010] Lee, U., Wang, P., Noh, Y., Vieira, L. F. M., Gerla, M., and Cui, J.-H. (2010). Pressure routing for underwater sensor networks. In *INFOCOM*, San Diego, CA.

[Li et al. 2000] Li, J., Jannotti, J., Couto, D. S. J. D., Karger, D. R., and Morris, R. (2000). A Scalable Location Service for Geographic Ad Hoc Routing. In *MOBICOM'00*, Boston, MA.

[Liu et al. 2005] Liu, B., Brass, P., Dousse, O., Nain, P., and Towsley, D. (2005). Mobility improves coverage of sensor networks. In *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 300–308, New York, NY, USA. ACM Press.

[Lougheed and Clifton 1988] Lougheed, J. and Clifton, R. (1988). A design for a waterside security system. In *Security Technology*, pages 39–43.

[Lovik et al. 2007] Lovik, A., Bakken, A., Dybedal, J., Knudsen, T., and Kjoll, J. (2007). Underwater protection system. pages 1 –8.

[Molins 2006] Molins, M. (2006). Slotted fama: a mac protocol for underwater acoustic networks. In *In IEEE OCEANS 06, Sigapore*, pages 16–19.

[N. Chirdchoo, W.-S. Soh, and K.-C. Chua 2008] N. Chirdchoo, W.-S. Soh, and K.-C. Chua (2008). RIPT: A receiver-initiated reservation-based protocol for underwater acoustic networks. volume 26, pages 1744–1753.

[Ottino 1989] Ottino, J. M. (1989). *The Kinematics of Mixing: Stretching, Chaos, and Transport*. Number 3 in Cambridge Texts in Applied Mathematics. Cambridge University Press.

[Owen et al. 2005] Owen, R., Mitchelmore, C., Woodley, C., Trapido-Rosenthal, H., Galloway, T., Depledge, M., Readman, J., Buxton, L., Sarkis, S., Jones, R., , and Knap., A. (2005). A common sense approach for confronting coral reef decline associated with human activities. In *Marine Pollution Bulletin*, volume 51, pages 481–485.

[Pahlavan and Levesque] Pahlavan, K. and Levesque, A. H. Wireless information networks.

[Partan et al. 2006] Partan, J., Kurose, J., and Levine, B. N. (2006). A survey of practical issues in underwater networks. In *WUWNet'06*, pages 17–24, Los Angeles, CA, USA.

[Pedlosky 1996] Pedlosky, J. (1996). *Ocean Circulation Theory*. Springer-Verlag, Heidelberg.

[Perkins 1997] Perkins, C. (1997). Ad hoc on demand distance vector (aodv) routing.

[Perkins and Bhagwat 1994] Perkins, C. E. and Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. *SIGCOMM Comput. Commun. Rev.*, 24(4):234–244.

[Pompili et al. 2006] Pompili, D., Melodia, T., and Akyildiz, I. F. (2006). Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks. In *MobiCom '06*, pages 298–309, Los Angeles, CA, USA.

[Pompili et al. 2009] Pompili, D., Melodia, T., and Akyildiz, I. F. (2009). A cdma-based medium access control for underwater acoustic sensor networks. *Trans. Wireless. Comm.*, 8(4):1899–1909.

[Preisig 2007] Preisig, J. (2007). Acoustic propagation considerations for underwater acoustic communications network development. *SIGMOBILE Mob. Comput. Commun. Rev.*, 11(4):2–10.

[Provenzale 1999] Provenzale, A. (1999). Transport by coherent barotropic vortices. *Annual Rev. Fluid Mech.*, 31:55–93.

[Rappaport 1983] Rappaport, T. (1983). *Principles of Underwater Sound*.

[Rappaport 2002] Rappaport, T. (2002). *Wireless Communications: Principles and Practice*.

[Ratnasamy et al. 2003] Ratnasamy, S., Karp, B., Shenker, S., Estrin, D., Govindan, R., Yin, L., and Yu, F. (2003). Data-Centric Storage in Sensornets with GHT, A Geographic Hash Table. *Springer Mobile Networks and Applications*, 8(4):427–442.

[Roberts 1975] Roberts, L. G. (1975). Aloha packet system with and without slots and capture. *SIGCOMM Comput. Commun. Rev.*, 5(2):28–42.

[Rossby et al. 1986] Rossby, T., Dorson, D., and Fontaine, J. (1986). The rafos system. *J. Atmos. Oceanic Tech.*, 3(4):672–679.

[Rubin and Ping Wu 2000] Rubin, S. and Ping Wu, H. (2000). A novel fiber-optic sensor for the long-term, autonomous measurement of pco2 in seawater. volume 1, pages 631 –639 vol.1.

[Samelson 1992] Samelson, R. M. (1992). Fluid exchange across a meandering jet. *J. Phys. Ocean.*, 22(4):431–440.

[Samelson and Wiggins 2006] Samelson, R. M. and Wiggins, S. (2006). *Lagrangian Transport in Geophysical Jets and Waves. The Dynamical Systems Approach*. Number 31 in Interdisciplinary Applied Mathematics. Springer-Verlag.

[Sarkar et al. 2006] Sarkar, R., Zhu, X., and Gao, J. (2006). Double Rulings for Information Brokerage in Sensor Networks. In *MOBICOM'06*, Los Angeles, CA.

[Savvides et al. 2002] Savvides, A., Park, H., and Srivastava, M. B. (2002). The bits and flops of the n-hop multilateration primitive for node localization problems. In *Proc of. WSNA '02*, pages 112–121, Atlanta, Georgia, USA.

[Smith et al. 1997] Smith, S., Park, J., and Neel, A. (1997). A peer-to-peer communication protocol for underwater acoustic communication. In *Oceans*, pages 268–272.

[Smookler et al. 2005] Smookler, M., Clark, B., and Ostrander, J. (2005). Underwater detection and surveillance technology for commercial port and vessel security. who is going to pay for it? In *Oceans*, pages 935–940.

[Sozer et al. 2000] Sozer, E., Stojanovic, M., and Proakis, J. (2000). Underwater acoustic sensor networks. *IEEE Journal of Oceanic Engineering*, 25(2):72–83.

[Stojanovic 1996] Stojanovic, M. (1996). Recent Advances in High-speed Underwater Acoustic Communications. In *IEEE Journal of Oceanic Engineering*, volume 21, pages 125–136.

[Stojanovic 2006] Stojanovic, M. (2006). On the relationship between capacity and distance in an underwater acoustic communication channel. In *WUWNet '06: Proceedings of the 1st ACM international workshop on Underwater networks*, pages 41–47, Los Angeles, CA, USA. ACM.

[Stojmenovic et al. 2006] Stojmenovic, I., Liu, D., and Jia, X. (2006). A Scalable Quorum based Location Service in Ad Hoc and Sensor Networks. In *MASS'06*, Vancouver, Canada.

[Sybrandy and Niiler 1991] Sybrandy, A. and Niiler, P. (1991). Woce/toga lagrangian drifter construction manual. SIO REF 91/6, WOCE Report 63.

[Takagi and Kleinrock 1987] Takagi, H. and Kleinrock, L. (1987). Correction to throughput analysis for persistent csma systems. In *Communications, IEEE Transactions on*, volume 35, pages 243–245.

[Tobagi and Kleinrock 1975] Tobagi, F. and Kleinrock, L. (1975). Packet switching in radio channels: Part ii the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. In *Communications, IEEE Transactions on*, volume 23, pages 1417–1433.

[Vieira 2009] Vieira, L. F. M. (2009). *Underwater SEA Swarm*. PhD thesis, University of California, Los Angeles.

[Vieira et al. 2006] Vieira, L. F. M., Kong, J., Lee, U., and Gerla, M. (2006). Analysis of Aloha Protocols for Underwater. In *ACM International Workshop on Underwater Networks, WUWNet'06*, Los Angeles, CA.

[Vieira et al. 2008] Vieira, L. F. M., Lee, U., and Gerla, M. (2008). Phero-Trail: a Bio-inspired Location Service for Mobile Underwater Sensors. In *ACM International Workshop on Underwater Networks, WUWNet'08*, San Francisco, CA.

[Vieira et al. 2009] Vieira, L. F. M., Lee, U., and Gerla, Y. N. M. (2009). LPS: Laser Positioning System for Underwater Networks. In *ACM International Workshop on Underwater Networks, WUWNet'09*, Berkeley, CA.

[WUWNET] WUWNET. 2009 program `http://wuwnet.acm.org/2009/program.php`.

[Xie and Cui 2007] Xie, P. and Cui, J.-H. (2007). R-mac: An energy-efficient mac protocol for underwater sensor networks. In *WASA '07: Proceedings of the International Conference on Wireless Algorithms,Systems and Applications*, pages 187–198, Washington, DC, USA. IEEE Computer Society.

[Yackoski and Shen 2008] Yackoski, J. and Shen, C.-C. (2008). Uw-flashr: achieving high channel utilization in a time-based acoustic mac protocol. In *WuWNeT '08: Proceedings of the third ACM international workshop on Underwater Networks*, pages 59–66, New York, NY, USA. ACM.

[Yan et al. 2008] Yan, H., Shi, Z. J., and Cui, J.-H. (2008). Dbr: Depth-based routing for underwater sensor networks. In *Networking*, Lecture Notes in Computer Science, pages 72–86. Springer.

[Yu et al. 2004] Yu, Y., Lu, G.-H., and Zhang, Z.-L. (2004). Enhancing Location Service Scalability with HIGH-GRADE. In *MASS'04*, Fort Lauderdale, FL.