
Universidade Federal do Ceará
Centro de Ciências
Departamento de Computação
Mestrado e Doutorado em Ciência da Computação

# INCENTIVANDO A COOPERAÇÃO EM REDES AD HOC

Marcos Dantas Ortiz

DISSERTAÇÃO DE MESTRADO

Fortaleza-CE
2007

Universidade Federal do Ceará
Centro de Ciências
Departamento de Computação

Marcos Dantas Ortiz

# INCENTIVANDO A COOPERAÇÃO EM REDES AD HOC

Dissertação apresentada ao Programa de Mestrado e Doutorado em Ciência da Computação do Departamento de Computação da Universidade Federal do Ceará como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. José Neuman de Souza, Dr
Co-orientador: Prof. Marcial Porto Fernandez, DSc

Fortaleza-CE
2007

**Incentivando a Cooperação em Redes Ad Hoc**

Marcos Dantas Ortiz

Dissertação submetida a Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Ceará como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

---

Prof. José Neuman de Souza, Dr
Universidade Federal do Ceará

---

Prof. Marcial Porto Fernandez, DSc
Universidade Estadual do Ceará

---

Profa. Rossana Andrade, PhD
Universidade Federal do Ceará

---

Prof. José Ferreira de Rezende, Dr
Universidade Federal do Rio de Janeiro

*Aos meus pais, Sergio e Mirzia ($in\ memorian$).*
*À minha irmã Mariana.*
*Ao meu avô Dantas ($in\ memorian$).*
*À minha amada Diana.*

# AGRADECIMENTOS

Agradeço primeiramente a Deus, que me dá saúde, amor e confiança para seguir na caminhada em busca dos meus objetivos.

Aos meus pais, Sergio e Mirzia (*in memorian*), pelo amor, atenção, exemplo de pessoa e educação. Desde cedo mostraram a importância do estudo e da dedicação. À minha irmã Mariana, pelo incentivo, confiança e torcida sempre depositados em mim. Ao meu avô Dantas (*in memorian*), pelo companheirismo e dedicação prestados na forma de um segundo pai, sempre valorizando minhas conquistas. Aos meus avós Nirce e Plácido, que, apesar da distância, acompanham e torcem muito por mim. A toda minha família que sempre apoiou e acreditou neste *estudante profissional*. À minha amada Diana, pelo amor, carinho, e dedicação prestados desde a época da graduação.

Ao professor Neuman pelo apoio, orientação, amizade e incentivo. Ao professor Marcial pela orientação, ajuda, motivação e confiança. À professora Rossana pela infraestrutura utilizada na elaboração deste trabalho e apoio financeiro. Ao professor Aldri pelos ensinamentos e atenção dispensada. Aos revisores do SBRC que fizeram considerações importantes à estratégia proposta neste trabalho.

Aos colegas da época do Colégio Santo Inácio que compreenderam a minha ausência nesses dois anos de mestrado. Aos colegas da UECE, *OS NERDS*, pelo companheirismo e atenção prestados após o término da graduação. Agradeço ao Acélio, aluno de Iniciação Científica da UECE, pela dedicação e ajuda na elaboração dos testes apresentados neste trabalho. Aos novos colegas conquistados após minha entrada no GREat. Ao Orleyzin pelo bom humor e atenção.

Aos desenvolvedores das ferramentas (LaTeX, Eclipse, NS-2, BonnMotion, ConfInt e Gnuplot) utilizadas no desenvolvimento deste trabalho.

Por fim, à FUNCAP que possibilitou minha dedicação exclusiva na elaboração deste trabalho

*E sem saber que era impossível, ele foi lá e fez*

—JEAN COCTEAU

# RESUMO

Redes móveis Ad Hoc (Mobile Ad Hoc Network - MANET) são redes sem fio que não possuem infra-estrutura de comunicação fixa e administração centralizada. Dessa forma, o gerenciamento da comunicação é realizado pelos próprios dispositivos, que, além de transmitir e receber pacotes, podem funcionar como roteadores. Na função de roteador, os dispositivos executam um papel fundamental para manutenção da conectividade da rede, o encaminhamento de pacotes. Devido ao pequeno alcance dos rádios de transmissão, nem sempre é possível a comunicação direta entre os dispositivos origem e destino. Dessa forma, grande parte das comunicações é realizada através do encaminhamento de pacotes por múltiplos saltos. Os dispositivos que realizam o encaminhamento de pacotes são chamados de nós intermediários. Os protocolos de roteamento desenvolvidos para MANETs assumem que o encaminhamento cooperativo de pacotes é sempre vantajoso para os dispositivos, no entanto, é sabido que na fase de transmissão de um pacote é verificado o maior consumo de recursos (e.g., processamento e energia). Nas MANETs de uso civil, em que os objetivos e as métricas de desempenho dos integrantes são diversos, os dispositivos podem apresentar um comportamento egoísta, através do não encaminhamento de pacotes, como forma de economizar seus recursos. Este trabalho apresenta uma estratégia autônoma de incentivo à cooperação baseada no uso de créditos. Os dispositivos que encaminham pacotes, recebem em troca créditos, enquanto os dispositivos que originam pacotes, perdem créditos. Como incentivo, o encaminhamento prioritário de pacotes fornece uma melhor Qualidade de Serviço (QoS) aos nós que possuem créditos. A estratégia utiliza apenas informações locais e não há necessidade de uma entidade central para gerenciar os créditos dos nós. Através de simulações, foram realizados vários experimentos em que foi verificada a eficácia desta estratégia no encaminhamento prioritário dos pacotes gerados pelos nós cooperativos.

**Palavras-chave:** Redes ad hoc, Cooperação, QoS, Encaminhamento prioritário

# ABSTRACT

Mobile Ad Hoc Networks (MANETs) refer to the wireless networks without a fixed infrastructure and a central administration. In this way, the communication management is self-organized by the mobile nodes in the network. The mobile nodes in MANETs can have a router role. These nodes are known as intermediate nodes and they have an important role to maintain the network connectivity forwarding packets for the benefit of other nodes. Because the small range of the radio transmission, most of the mobile nodes (sender and receiver) cannot usually communicate with each other directly. So the communications are relying on multihop routing. Routing protocols for MANETs assume that the nodes are willing to participate in the cooperative packets forwarding, however, in the packet transmission stage it is verified a great resource consumption (e.g., processing and energy). Since forwarding a packet will incur a cost of energy to a node, the nodes can have a selfish behavior avoiding send packets to save their resources. This work describes an autonomous strategy to provide an incentive for mobile nodes to cooperate in a credit-based system. A node receives credit for forwarding a packet while the sender node looses credit. As an incentive for mobile nodes to cooperate they receive a better quality of service (QoS). This strategy uses the amount of credits to classify the flows inside routers nodes. This strategy uses only local information and it does not have a central entity to manage the credit of the nodes. The simulations show that this strategy is effective to prioritize packets sent by cooperative nodes.

**Keywords:** Ad-hoc Networks, Cooperation and QoS

# SUMÁRIO

# LISTA DE ABREVIATURAS

**CCS** - Credit Clearance Service

**CSMA/CA** - Carrier Sense Multiple Access with Collision Avoidance

**DCF** - Distributed Coordination Function

**DoS** - Denial of Service

**EN** - Equilíbrio de Nash

**FIFO** - First In First Out

**MAC** - Media Access Control

**MANET** - Mobile Ad Hoc Network

**NS-2** - Network Simulator Version 2.0

**P2P** - Peer-to-Peer

**PA** - Ponto de Acesso

**PAN** - Personal Area Networks

**PPM** - Packet Purse Model

**PTM** - Packet Trade Model

**QoS** - Quality of Service

**RERR** - Route Error

**RREP** - Route Replay

**RREQ** - Route Request

**SDI** - Sistema de Detecção de Intrusão

**TCL** - Tabela de Créditos Local

**TCP** - Transmission Control Protocol

**TFT** - Tit-for-Tat

**UDP** - User Datagram Protocol

**WLAN** - Wireless Local Area Network

**WRR** - Weighted Round-Robin

# LISTA DE FIGURAS

# LISTA DE TABELAS

# CAPÍTULO 1

# INTRODUÇÃO

Redes móveis ad hoc (*Mobile Ad Hoc Networks* – MANETs ) são redes sem fio, formadas por uma coleção de dispositivos móveis[1] autônomos capazes de se auto-organizarem [1]. Como a movimentação dos nós é livre, a rede possui uma topologia dinâmica e arbitrária. Esse tipo de rede é caracterizado pela ausência de uma infra-estrutura de comunicação, mobilidade, dinamismo e a necessidade de cooperação entre os dispositivos móveis.

O crescente uso de redes *ad hoc* como alternativa às redes cabeadas e às redes sem fio infra-estruturadas tem fomentado um grande número de pesquisas, cujo foco principal é o aprimoramento do comportamento autônomo dos dispositivos. Devido à ausência de uma infra-estrutura de rede, a gerência das comunicações é de responsabilidade dos próprios nós. A comunicação entre os dispositivos móveis pode ser estabelecida de duas formas distintas: direta, quando estão dentro raio de transmissão; ou indireta, através de encaminhamento por múltiplos saltos.

Na comunicação direta, os nós envolvidos não dependem de auxílio para estabelecer a comunicação, entretanto, na comunicação por múltiplos saltos, outros nós, chamados intermediários, são responsáveis pelo estabelecimento da comunicação. Os nós intermediários funcionam como roteadores, o que permite uma transmissão por múltiplos saltos de pacotes entre origens e destinos distantes (o raio de transmissão não é suficiente para alcançar o nó destino). Portanto, os nós de uma rede *ad hoc* funcionam como estações terminais e roteadores. O papel de roteador dos nós é fundamental para a manutenção da conectividade da rede, visto que a forma de estabelecimento de comunicação mais freqüente é através de múltiplos saltos.

## 1.1 CARACTERIZAÇÃO DO PROBLEMA E MOTIVAÇÃO

As redes *ad hoc* são utilizadas mais comumente para as seguintes aplicações: militar, de sensoriamento, de operações de resgate, de substituição de uma infra-estrutura de rede

[1]Neste trabalho, os termos dispositivos móveis e nós são utilizados com a mesma conotação.

danificada por eventos naturais, e de uso civil [2]. Nas quatro primeiras aplicações, é comum a existência de uma autoridade[2] responsável pela coleção de dispositivos móveis [3]. Nas redes *ad hoc* de sensoriamento, por exemplo, os nós devem cooperar na divulgação das observações realizadas localmente. O tipo de sensoriamento pode até ser semelhante, o que reduz a heterogeneidade da rede [4]. Portanto, esses dispositivos são desenvolvidos para alcançar um objetivo comum definido pela autoridade.

Nas redes *ad hoc* civis, os nós podem apresentar um comportamento indesejado devido à perda de uma autoridade que os obrigue a encaminhar os pacotes de outros nós. Como na fase de transmissão de pacotes é verificado um maior consumo no uso de recursos (e.g., carga da bateria, ciclo do processador), os nós tendem a encaminhar uma pequena quantidade de pacotes na tentativa de economizar seus recursos, o que representa o comportamento egoísta, foco deste trabalho.

Outro tipo de comportamento indesejado é o malicioso. O nó malicioso não coopera com a rede apenas para economizar seus recursos, mas, principalmente, para prejudicar outros nós ou utilizar os recursos da rede com uma maior vantagem [5]. Dessa forma, os nós maliciosos causam problemas relacionados à segurança da rede. Neste trabalho, esse tipo de comportamento não é tratado.

Alguns protocolos de roteamento encontrados na literatura (e.g., AODV [6], DSDV [7], TORA [8] e DSR [9]) foram desenvolvidos para otimizar o processo de comunicação entre os nós e, até mesmo, evitar rotas que utilizem nós com recursos escassos. Entretanto, esses protocolos assumem que os nós são cooperativos no encaminhamento de pacotes. Jindal et al. [10] mostram que a existência de cooperação entre os nós pode aumentar significativamente o desempenho global de uma rede *ad hoc*. Portanto, faz-se necessário o desenvolvimento de soluções específicas que incentivem a cooperação.

Diversas abordagens têm sido propostas para evitar ou punir o comportamento indesejado dos nós de uma rede *ad hoc*. O comportamento indesejado, ou mau comportamento, ocorre quando um nó causa problemas relacionados à segurança da rede ou ao paradigma da cooperação. Os tipos de incentivos mais utilizados pelas abordagens são: baseados em reputação, baseados em créditos e baseados em teoria dos jogos.

Nas abordagens baseadas em reputação é proposto que cada nó avalia o com-

[2]Entidade (e.g., departamento militar, órgão de pesquisa) responsável por definir a configuração e os objetivos da rede. São definidos, por exemplo, quais os tipos de dispositivos da rede, mecanismo de controle de energia, protocolo de roteamento e tipo de endereçamento.

portamento de seus vizinhos através do uso do modo promíscuo[3]. Dessa forma, quando um nó recebe um pacote que deve ser encaminhado, seus vizinhos podem vigiá-lo. Caso um nó não encaminhe um pacote, ele está apresentando um comportamento indesejado para seus vizinhos. Repetindo esse comportamento um determinado número de vezes, é atribuída ao nó uma baixa reputação perante a rede. Para isso, as informações de reputação são difundidas pela rede ou enviadas para uma entidade central de análise de comportamento.

As abordagens baseadas em créditos determinam que um nó só pode enviar seus pacotes se possuir créditos suficientes para todo o trajeto de transmissão. Os nós que encaminham pacotes de outros nós recebem créditos em troca pelo serviço prestado. Dessa forma, os nós que não encaminham pacotes não conseguem adquirir os créditos necessários para o estabelecimento de suas comunicações.

Nas abordagens baseadas em teoria dos jogos, a rede *ad hoc* é modelada como um ambiente de jogo em que os nós são jogadores independentes. Nesse tipo de abordagem, também é utilizado o conceito de créditos como condição para o envio de pacotes. A cooperação é motivada por ser considerada, pelos jogadores, a melhor estratégia a ser seguida para alcançar uma grande quantidade de benefícios. Caso seja usada uma entidade central de gerenciamento de créditos, ela pode ser modelada como um outro jogador.

Algumas abordagens defendem a teoria de que a dependência mútua existente entre os nós de uma rede *ad hoc* seria suficiente para provocar um equilíbrio cooperativo sobre o comportamento dos nós. Dessa forma, pelo menos uma estratégia cooperativa é seguida pelos nós sem a necessidade de incentivá-los.

## 1.2 OBJETIVO

Este trabalho apresenta uma estratégia de incentivo à cooperação baseada em créditos que visa combater o comportamento egoísta. A participação dos nós no encaminhamento de pacotes é incentivada através da melhoria de qualidade de serviço (*Quality of Service* - QoS). O uso de boa QoS como incentivo à cooperação foi motivado pela baixa vazão apresentada pelas redes *ad hoc* [11], logo, o nó que possui uma quantidade positiva de créditos é beneficiado através de uma melhor QoS para seus fluxos, enquanto os nós com

[3]Cada nó escuta os pacotes transmitidos pelos seus vizinhos mesmo quando os pacotes não são endereçados para ele.

créditos negativos são punidos pelo mecanismo de encaminhamento prioritário de pacotes. A análise do comportamento dos nós é realizada através de um sistema de créditos que contabiliza o encaminhamento dos pacotes.

A estratégia proposta utiliza apenas informações locais e independe de uma entidade central para gerenciar os benefícios. Nas redes *ad hoc* civis, o uso de uma entidade central de gerenciamento não é comum, visto que essas redes não são desenvolvidas e gerenciadas por uma autoridade.

## 1.3 CONTRIBUIÇÕES

O desenvolvimento e avaliação de uma estratégia local e descentralizada de incentivo à cooperação trazem as seguintes contribuições:

- Desenvolvimento e avaliação de um mecanismo voltado ao incentivo à cooperação em redes *ad hoc*. Foram desenvolvidas extensões ao simulador NS-2 (modificações no protocolo de roteamento DSR, desenvolvimento de um escalonador para as filas de envio e encaminhamento de pacotes, baseado na disciplina WRR, e a implementação do comportamento egoísta sobre os nós móveis) que permitiram simular o mecanismo proposto sobre uma rede *ad hoc*.
- Publicações referentes ao desenvolvimento do mecanismo proposto em [12], em que a necessidade de incentivo à cooperação é abordada e é proposta uma solução simples em que foram realizados testes iniciais, e em [13], em que foi apresentado uma estratégia mais elaborada para o mesmo problema e realizados testes sobre métricas de QoS.

## 1.4 ESTRUTURA DO TRABALHO

Este trabalho está organizado em cinco capítulos descritos a seguir. No Capítulo 1, apresentou-se o problema a ser tratado pelo mecanismo proposto, os objetivos a serem alcançados, o estado da arte de forma resumida e, por fim, as contribuições resultantes do desenvolvimento deste trabalho.

O Capítulo 2 apresenta uma visão geral sobre as principais características das redes *ad hoc*. Neste capítulo, são descritos o conceito de uma rede *ad hoc*, suas prin-

cipais características, os possíveis tipos de aplicações, os protocolos de roteamento e o provimento de QoS. O Capítulo 2 também aborda a importância da cooperação entre os dispositivos de uma rede *ad hoc*, apresenta os possíveis tipos de comportamento adotados pelos dispositivos móveis e, por fim, apresenta o estado da arte sobre os mecanismos que visam a incentivar a cooperação em redes *ad hoc*.

O Capítulo 3 descreve a estratégia de incentivo à cooperação proposta neste trabalho. Neste capítulo, é abordada uma visão geral sobre o mecanismo proposto, as suposições para uso do mecanismo proposto, os tipos e o escopo das informações utilizadas, o sistema de créditos e o mecanismo de priorização de encaminhamento.

O Capítulo 4 apresenta o ambiente de simulação, os tipos de testes realizados e os resultados obtidos. Neste capítulo também são descritas as modificações realizadas sobre o protocolo de roteamento DSR e sua fila de pacotes.

O Capítulo 5 apresenta as conclusões obtidas das simulações e implementações deste trabalho. Também são abordadas as possíveis extensões do mecanismo proposto e as investigações futuras relacionadas ao incentivo à cooperação adotado por este trabalho.

# CAPÍTULO 2

# REDES AD HOC

Este Capítulo apresenta uma visão geral sobre as principais características das redes *ad hoc* e aborda a importância de incentivar a cooperação entre os dispositivos móveis. Inicialmente, na Seção 2.1, é descrito o conceito de uma rede *ad hoc* e suas principais características. Na Seção 2.2 são apresentadas as novas características introduzidas pelas redes *ad hoc*. A Seção 2.3 aborda os possíveis tipos de aplicações para redes *ad hoc*. Na Seção 2.4 são descritos os tipos de protocolos de roteamento com ênfase no protocolo reativo DSR. A Seção 2.5 aborda, resumidamente, o provimento de QoS em redes *ad hoc*. A Seção 2.6 aborda a importância da cooperação entre os dispositivos de uma rede *ad hoc*, apresenta os possíveis tipos de comportamento executados pelos dispositivos móveis e, por fim, o estado da arte sobre os mecanismos que visam incentivar a cooperação em redes *ad hoc*.

## 2.1 CONCEITO

Atualmente, os principais padrões de comunicação sem fio com suporte à MANET, são: o IEEE 802.11 [14], utilizado também na criação de redes locais sem fio (*Wireless Local Area Networks* - WLANs), e o *Bluetooth* [15], utilizado na criação de redes pessoais sem fio (*Personal Area Networks* - PAN).

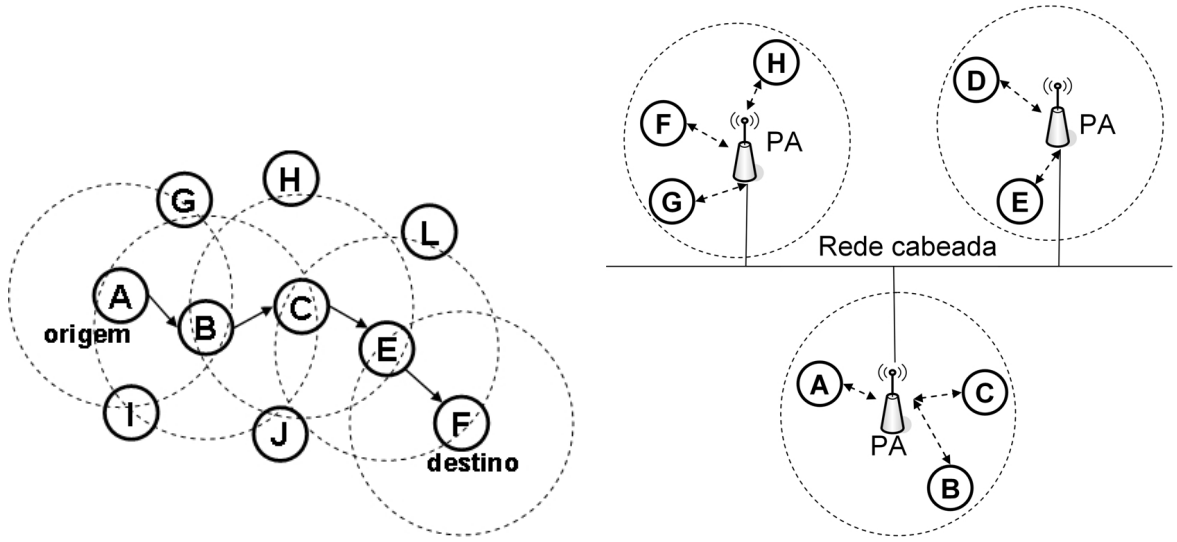
Uma MANET possui como principal característica a ausência de uma infra-estrutura de comunicação e administração. Nas redes sem fio infra-estruturadas, toda comunicação entre os nós é gerenciada por uma entidade centralizadora, chamada de Ponto de Acesso (PA). Já nas redes *ad hoc* os nós se comunicam sem a necessidade de uma entidade central de administração.

Cada dispositivo é equipado com um rádio transmissor e um receptor, o que permite a comunicação sem fio com outros dispositivos. Portanto, os nós são capazes de gerar/receber pacotes para/de os outros nós da rede. Os nós também podem atuar

no encaminhamento de pacotes de outros nós executando a função de roteadores. Nesta função, são chamados de nós intermediários da comunicação.

A comunicação através de nós intermediários é necessária quando o raio de alcance da transmissão do nó origem não alcança o nó destino e vice-versa. Neste caso, o nó origem necessita de uma rota através de múltiplos saltos até o nó destino. Dessa forma, o alcance de sinal é virtualmente estendido para possibilitar a comunicação. O roteamento através de múltiplos saltos é extremamente utilizado nas MANETs devido ao curto alcance de transmissão.

A Figura 2.1(a) ilustra um exemplo de uma MANET, em que é utilizado encaminhamento por múltiplos saltos para possibilitar a comunicação entre os nós $A$ e $F$. A rota seguida pelos pacotes é representada por setas e os círculos representam o alcance do sinal de cada nó. Na Figura 2.1(b) é ilustrado uma rede sem fio infra-estruturada gerenciada por três PAs que estão conectados à rede cabeada local. As setas representam a comunicação entre os nós e o respectivo PA responsável, enquanto as circunferências representam a área de atuação dos PAs.




(a) MANET (b) WLAN

**Figura 2.1** Redes sem fio

## 2.2 PRINCIPAIS CARACTERÍSTICAS

As MANETs herdaram muitas características das redes sem fio e cabeadas, porém, adicionaram outras novas. Essas características representam tanto desafios quanto oportunidades para o desenvolvimento de novas aplicações, que não eram possíveis ou seriam mais custosas se utilizadas redes sem fio infra-estruturadas. Com base nos trabalhos de Basagni [2] e Hekmat [16], as principais características das redes *ad hoc* são apresentadas a seguir:

- **Comunicação sem fio:** o uso de fios não é necessário para o estabelecimento das comunicações entre os dispositivos.

- **Mobilidade:** os nós são livres para se locomoverem durante sua permanência na rede, mesmo quando estão se comunicando com outros nós. Portanto, a topologia das MANETs é dinâmica e imprevisível.

- **Ausência de infra-estrutura de comunicação:** a comunicação entre os dispositivos independe de uma entidade central de administração. Essa característica ressalta a capacidade autônoma desses dispositivos que são capazes de se auto-organizarem.

- **Auto-organização:** a MANET é capaz de decidir sua configuração de forma autônoma, através da configuração de parâmetros como: o tipo de endereçamento utilizado, o protocolo de roteamento, o mecanismo de localização,o mecanismo de economia de energia, formação de *cluster*, mecanismo de segurança, mecanismo de incentivo ao comportamento cooperativo, dentre outros.

- **Roteamento através de múltiplos saltos:** as MANETs não dependem de roteadores fixos para o estabelecimento das comunicações. Os dispositivos são capazes de encaminhar pacotes de outros nós, o que possibilita a manutenção da conectividade da rede. O roteamento é realizado através de múltiplos saltos entre os nós intermediários da comunicação.

- **Energia limitada:** a maioria dos dispositivos móveis (e.g., *notebooks*, *PDAs*, sensores e celulares) possuem baterias com baixa autonomia e, geralmente, não são capazes de gerar sua própria energia. Portanto, os protocolos e aplicações desenvolvidos para esses dispositivos visam minimizar o consumo de energia.

## 2.3 APLICAÇÕES

O desenvolvimento de MANETs possibilitou o surgimento de novas aplicações que não eram possíveis sem as novas funcionalidades introduzidas por essas redes. A flexibilidade das MANETs permite a construção de redes em qualquer lugar, a qualquer tempo e entre dispositivos heterogêneos. A independência de uma entidade central de gerenciamento e administração permite a formação de redes onde existe pouca ou nenhuma infra-estrutura de comunicação, o que reduz o custo de sua implantação. A capacidade de auto-organização e a mobilidade presentes nas MANETs possibilitam a criação de aplicações dinâmicas e independentes.

As primeiras aplicações para redes *ad hoc* eram voltadas para uso militar [17]. O dinamismo das operações militares e a impossibilidade de construção de uma infra-estrutura de comunicação durante essas operações, impulsionaram o desenvolvimento das redes *ad hoc*. Portanto, as MANETs permitiram, inicialmente, a comunicação autônoma entre tropas e dispositivos de guerra.

Outro exemplo da aplicação de redes *ad hoc* são as redes de sensores sem fio. As redes de sensores são largamente utilizadas na coleta de dados em tempo real [18]. Os elementos formadores dessas redes possuem sensores embarcados dedicados à observação, por exemplo, de fenômenos da natureza (e.g., terremotos, queimadas e emissão de gases), monitoração do corpo humano (e.g., sinais vitais, temperatura e nível de glicose no sangue) e em ambientes de guerra. Os dados coletados são enviados por inundação ou múltiplos saltos às entidades de processamento responsáveis.

Os sensores podem ser distribuídos em diversos tipos de ambientes, desde os inóspitos aos seres humanos, onde podem formar redes muito densas com centenas ou milhares de dispositivos [19], ou, até mesmo, no próprio corpo humano [20]. Essas redes possuem limitações severas de energia, processamento e memória, portanto, muito esforço tem sido dado ao desenvolvimento de mecanismos de economia de energia.

Redes *ad hoc* também são utilizadas em operações de resgate, nas quais as equipes

de resgate podem trocar informações em ambientes desprovidos de infra-estrutura de comunicação. Os dispositivos também são utilizados na coleta de dados em ambientes de difícil acesso. Essas redes são utilizadas na substituição temporária de uma infra-estrutura fixa de comunicação, danificada por eventuais fenômenos da natureza (e.g., terremotos, furacões e incêndios) [16].

Nos últimos anos, observa-se o desenvolvimento de aplicações para o uso civil. O dinamismo, a flexibilidade e a mobilidade oferecidos pelas MANETs motivam o desenvolvimento de aplicações ubíquas, voltadas para o uso comercial, de entretenimento, veicular, doméstico, de aplicações *Peer-to-Peer* P2P (compartilhamento de serviços) e extensão do raio de alcance em redes WLAN, MESH e celulares [21]. Os dispositivos que compõem essas redes são utilizados no cotidiano das pessoas (e.g., *notebooks*, *PDAs* e celulares).

As aplicações para as redes *ad hoc* têm evoluído do uso militar para o uso civil, com o objetivo de facilitar o cotidiano e a comunicação entre as pessoas. Entretanto, o fato das redes *ad hoc* serem utilizadas por pessoas comuns com interesses diferentes, impõe alguns obstáculos à manutenção da conectividade, principalmente, pelo fato da comunicação ser realizada através de múltiplos saltos.

Diferentemente dos outros tipos de redes *ad hoc*, as redes civis não são gerenciadas por uma autoridade [3]. Dessa forma, tem-se desprendido um grande esforço no desenvolvimento de mecanismos de segurança e de incentivo à cooperação, que evitem, respectivamente, problemas de conectividade e de segurança na comunicação.

**Tabela 2.1** Aplicações para MANETs

| Rede | Aplicação | Autoridade |
|---|---|---|
| Militar | Comunicação no campo de guerra | X |
| Sensoriamento | Monitoramento e coleta de dados | X |
| Resgate | Comunicação no ambiente de resgate e coleta de dados | X |
| Substituição Temporária | Substituir uma infra-estrutura de comunicação danificada | X |
| Civil | Comercial, entretenimento, veicular e doméstico | - |

A Tabela 2.1 resume os tipos de redes *ad hoc* e suas aplicações. Com base no trabalho de Buttyan e Hubaux [3], esta tabela também indica para quais tipos de redes

*ad hoc* existe ou não uma autoridade responsável. Vale ressaltar que podem existir redes *ad hoc* formadas por dispositivos pertencentes a autoridades diferentes, como a sobreposição de redes de sensores que monitoram espécies diferentes de animais em um mesmo ambiente. Nesse caso, a cooperação pode ser incentivada entre domínios como foi proposto por Felegyhazi et al. em [22] e por Buttyan et al. em [23].

## 2.4 ROTEAMENTO

Os protocolos de roteamento são responsáveis pela criação das rotas de comunicação, as quais são formadas por saltos (nós intermediários) que possibilitam a comunicação quando a transmissão direta não é possível. Portanto, os protocolos de roteamento possuem um papel fundamental para redes *ad hoc*: possibilitar a comunicação e a disponibilização de serviços [4].

O desenvolvimento dos protocolos de roteamento ainda é considerado um grande desafio de pesquisa. A mobilidade presente nas MANETs dificulta a manutenção do estado global da rede. Dessa forma, na tentativa de prover uma rota satisfatória, os algoritmos de roteamento devem analisar características importantes presentes nessas redes (e.g., mobilidade, alta taxa de erros na transmissão e limitações de energia, banda e processamento).

Os algoritmos de roteamento devem ser robustos ao ponto de perceberem a ocorrência de mudanças na topologia da rede. O dinamismo da topologia pode provocar problemas na comunicação relacionados à queda da rota. Detectado o problema, o algoritmo cria rotas alternativas que visam minimizar esses problemas e, em alguns casos, otimizar o uso dos recursos.

Muitos algoritmos de roteamento têm sido desenvolvidos nos últimos anos. A forma mais comum de categorizá-los é através da análise de como a informação de roteamento é capturada e como as rotas são mantidas. As categorias mais comuns são: protocolos pró-ativos, reativos, híbridos e baseados em difusão.

Os protocolos baseados em difusão inundam a rede para possibilitar o envio dos pacotes. Na forma mais simples desse protocolo, o dispositivo envia cada pacote recebido aos seus vizinhos [24]. Protocolos baseados em difusão são úteis em situações em que a mobilidade dos nós é muito alta. Dessa forma, o uso de protocolos mais sofisticados poderia aumentar o processamento dos nós e sobrecarregar a rede com mensagens de

controle e, mesmo assim, não obter sucesso. Em geral, protocolos desse tipo apresentam grande uso de banda e consumo de energia dos dispositivos.

Os protocolos pró-ativos têm sua origem nos protocolos de redes cabeadas, porém o dinamismo das MANETs exige que esses protocolos se adaptem à ocorrência de mudanças freqüentes na topologia da rede. Para isso, os protocolos desse tipo devem manter uma visão consistente da rede. Cada nó mantém uma ou mais tabelas com informações de roteamento. Para manter a consistência das informações, os nós devem enviar notificações de atualização a cada mudança percebida na topologia da rede. Essas informações são utilizadas na formação de rotas para todos os destinos alcançáveis da rede, mesmo que não haja tráfego nas rotas. A desvantagem dessa categoria de protocolo é a sobrecarga (e.g., processamento, armazenamento e banda) gerada para manter as informações de topologia consistentes.

Os protocolos reativos são aqueles em que as rotas são construídas somente quando solicitadas pelos nós. Antes de iniciar uma conexão, o nó origem verifica se possui uma rota para o nó destino. Caso possua uma rota, a comunicação é iniciada, entretanto, quando isso não ocorre, o nó origem solicita ao protocolo de roteamento a construção (descoberta) de uma rota até o nó destino. O processo de descoberta de rotas é finalizado quando o nó destino é alcançado ou quando todas as permutações são analisadas. Uma vez que a rota é estabelecida, ela é mantida até que o nó destino não seja mais alcançado por nenhuma das rotas descobertas.

Devido à mobilidade dos nós observada nas redes civis, o uso de protocolos reativos é aconselhado, principalmente, por evitar a sobrecarga das mensagens de atualização de topologia e economizar recursos importantes como processamento, memória e energia. Neste trabalho, foi escolhido o uso do protocolo de roteamento *Dynamic Source Routing* (DSR) que é abordado a seguir.

### 2.4.1 Dynamic Source Routing (DSR)

O protocolo de roteamento DSR é do tipo reativo, voltado para criação de rotas por demanda do nó origem e designado para redes com topologias com alto dinamismo [9]. No DSR, cada nó participante da rede mantém uma *cache* de rotas, em que são armazenadas as rotas para os destinos desejados.

Quando um nó deseja se comunicar, porém, não possui uma rota para o destino

desejado, ele inicia o processo de descoberta de rotas (*Route Discovery*). Este processo é dinâmico e visa à formação de uma rota da origem até o destino.

O processo de descoberta de rotas é baseado em difusão, portanto, os pacotes de requisição de rota (*Route Request* - RREQ) são propagados através da rede com as seguintes informações: endereço do nó origem, endereço do nó destino, identificador do pacote de requisição e um registro contendo os nós visitados pelo pacote. O pacote RREQ é assinado com um identificador único para transmissões repetidas de um mesmo pacote. O processo seguido por um nó ao receber um pacote RREQ é descrito seguir:

1. Se a tupla (endereço do nó origem, identificador do pacote de requisição) estiver contido em alguma requisição recente, o pacote RREQ é descartado e nenhum processo adicional é executado.

2. Se o endereço do nó corrente estiver no registro de nós já visitados, o pacote RREQ é descartado e nenhum processo adicional é executado.

3. Se o nó corrente é o destino requisitado, um pacote de resposta à requisição de rota (*Route Reply* - RREP) é enviado ao nó fonte, iniciador do processo, contendo os nós, saltos, formadores da rota.

4. Se a rota para o destino estiver na sua *cache* de rotas, o nó adiciona essa rota ao registro de nós visitados e envia ao nó origem o pacote RREP contendo os nós formadores da rota.

5. Caso a rota não esteja em sua *cache*, o nó adiciona seu endereço ao registro de nós visitados e difunde o RREQ atualizado para seus vizinhos.

Quando o nó destino recebe um pacote RREQ, ele pode enviar um RREP ao nó origem de três formas diferentes. As formas de envio são determinadas pelo tipo de antena utilizada na transmissão. Se as antenas dos nós integrantes forem omnidirecionais e os enlaces bidirecionais, o pacote RREP pode ser enviado através da seqüência dos nós visitado pelo RREQ na ordem inversa. No caso das antenas serem direcionais, o processo de descoberta é utilizado novamente para encontrar a rota de volta ao nó origem. Uma variação desse processo é o conceito de carona (*Piggybacking*), em que as informações do RREP são armazenadas dentro do RREQ para otimizar o processo descoberta invertida. A diferença entre essas antenas é que na primeira os pacotes são enviados em todas as

direções dentro do raio de alcance, enquanto que na segunda os pacotes são enviados apenas dentro de um setor do raio, porém este raio é maior. A Figura 2.2 ilustra os tipos de transmissão de acordo com o tipo de antena, enquanto a Figura 2.3 ilustra o processo de descoberta de rotas em que são utilizadas antenas omnidirecionais.

Quando ocorre uma mudança na topologia (e.g., falha ou movimentação dos nós) suficiente para prejudicar alguma rota, o processo de manutenção de rotas é executado. Quando é detectado algum problema na rota, seja por falha ou movimentação de algum nó, um dos nós integrantes da rota, que percebeu a falha, envia um pacote notificando existência de erro nesta rota (*Route Error* - RERR). O pacote RERR contém o endereço do nó que detectou a falha e o endereço do nó que prejudicou a rota. Esse pacote é enviado ao nó origem para que seja executado outro processo de descoberta de rotas.

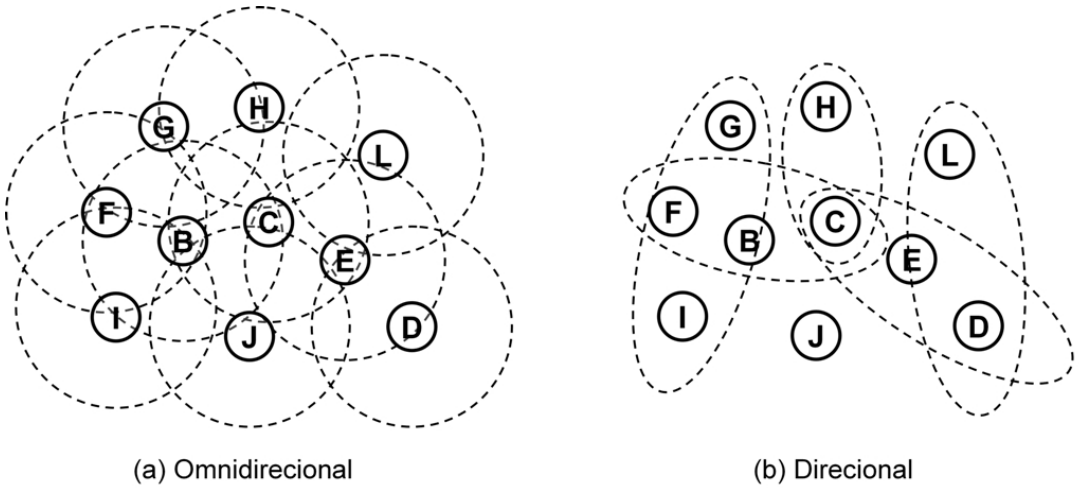



**Figura 2.2** Tipo de transmissão

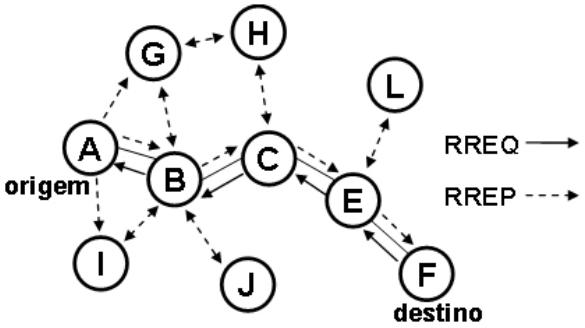



**Figura 2.3** Processo de descoberta de rotas

Diferentemente dos protocolos pró-ativos, o DSR não requer anúncios periódicos sobre a conectividade dos nós e mantém apenas as rotas requeridas para os destinos de-

sejados. Entretanto, os nós também podem incrementar suas informações sobre as rotas através da análise dos pacotes RREP difundidos, sem prejudicar a capacidade de armazenamento. O processo de manutenção de rotas pode ser otimizado se os nós mantiverem mais de uma rota para um mesmo destino, desde que não prejudique a escalabilidade da rede. As razões que motivaram o uso do protocolo DSR pelo mecanismo de incentivo à cooperação proposto neste trabalho são abordadas no Capítulo 3.

## 2.5 QUALIDADE DE SERVIÇO

Desde que a *Internet* foi disponibilizada para os usuários comuns, o tráfego gerado tem crescido de forma rápida. Inicialmente, era provido o melhor esforço para o tráfego dos usuários sem qualquer tipo de priorização. Atualmente, com o advento de aplicações que são sensíveis ao retardo e à largura de banda, estratégias têm sido propostas para permitir seu bom funcionamento, com algum tipo de priorização no uso dos serviços. O provimento de serviços pela rede com garantias de desempenho é o principal objetivo das estratégias de provisionamento de Qualidade de Serviço.

Nas MANETs, tem-se observado também o advento de um grande número de aplicações sensíveis aos serviços providos pela rede. Entretanto, a provisão de QoS torna-se um desafio significativamente maior que nas redes cabeados pela presença das seguintes características: mobilidade dos nós, limitações de banda, processamento, armazenamento e energia. A meta da provisão de QoS nessas redes é a boa distribuição de informações e o melhor aproveitamento possível dos recursos disponíveis [25].

Os serviços oferecidos, como garantia, pelo provisionador de QoS são os seguintes:

- **Largura de banda mínima:** capacidade mínima de banda passante oferecido ao usuário.
- **Retardo máximo:** tempo máximo que um pacote pode levar da origem até o destino.
- **Variação máxima do retardo:** também conhecido como *jitter*, indica o variação máxima no tempo de entrega.
- **Descarte máximo:** a taxa de descarte é obtida pelo número de pacotes que alcançam o destino divida pelo total de pacotes originados. É garantida uma taxa de descarte máxima.

Quando a rede aceita uma requisição de usuário, ela deve garantir que os requerimentos solicitados ao fluxo sejam observados durante a sua duração. Portanto, a rede deve prover garantias no provimento de serviços para os fluxos do usuário. Entretanto, o requerimento de QoS requer negociação entre o usuário solicitante e a rede. Essa negociação pode prover reserva de recursos, escalonamento prioritário de pacotes e controle de admissão, que podem ser direcionados por fluxo, por *link* ou por nó. O fato de não haver uma entidade central, torna essa negociação mais um desafio para o provisionamento de QoS em MANETs.

Reddy et al. [26] propõem uma classificação para as soluções existentes de QoS, as quais podem ser classificadas de acordo com a camada, da pilha de protocolos, na qual operam. São apresentadas três categorias de classificação: soluções que operam na camada MAC (*Media Access Control*), camada de rede e os que interagem com várias camadas (*frameworks* de QoS). A estratégia proposta neste trabalho opera na camada de rede para prover encaminhamento prioritário.

O uso de disciplinas de escalonamento sobre o encaminhamento de pacotes é um meio utilizado pelos *frameworks* desenvolvidos para operar na camada de rede, como o *INSIGNIA* [27], para determinar a ordem em que os pacotes pendentes são servidos. O *INSIGNIA* utiliza a disciplina de escalonamento *Weighted Round-Robin* (WRR) que utiliza informações locais sobre os canais de comunicação para classificar os fluxos. Uma alternativa para a classificação dos fluxos é a análise da quantidade de saltos pendentes até o destino [28]. Nesse caso, prioriza-se os pacotes que estão mais próximos do destino. Neste trabalho, a disciplina de escalonamento WRR é utilizada no escalonamento de pacotes, no entanto, os fluxos são classificados através da quantidade de créditos do nó origem.

Os protocolos de roteamento, que visam prover QoS, selecionam as rotas com recursos suficientes para satisfazer os requerimentos de cada fluxo. As informações relativas à disponibilidade dos recursos são gerenciadas com o objetivo de formar uma base de conhecimento que permita a escolha das rotas mais adequadas pelo algoritmo de roteamento. Alguns protocolos de roteamento, mais complexos, tentam otimizar o uso dos recursos.

O maior desafio desse tipo de protocolo é fornecer uma forma de gerenciar as informações sobre disponibilidade dos recursos. O algoritmo deve ser muito robusto para que esse gerenciamento não sobrecarregue a rede, o que poderia impossibilitar o suporte

às garantias oferecidas.

Neste trabalho, não são utilizadas informações sobre o estado dos enlaces nem recursos são reservados, no entanto, é provido o encaminhamento prioritário a cada salto. O mecanismo de provisionamento prioritário é detalhado no Capítulo 3.

## 2.6 COOPERAÇÃO

O termo cooperação significa trabalhar em conjunto visando somar resultados, aproveitando melhor os recursos disponíveis, em geral escassos. Há interesse especial quando a cooperação envolve grande número de participantes. A cooperação pode resultar em algo mais significativo do que qualquer um dos cooperantes poderia realizar sozinho [29].

A cooperação tem sido utilizada entre os seres humanos há muito tempo, entretanto, para esse tipo de indivíduo, a cooperação geralmente é motivada por incentivos. Cooperação não motivada, é aquela executada por indivíduos generosos, que auxiliam outros indivíduos sem receber recompensas. Este tipo de cooperação pode ser observado entre os integrantes de uma família.

A cooperação pode ser motivada através de incentivos individuais ou de grupo. Nos esportes coletivos (e.g., futebol, basquete e vôlei), os integrantes de um time cooperam na tentativa de conquistar a vitória (objetivo comum). Por outro lado, em alguns esportes individuais, como no ciclismo, atletas concorrentes cooperam em primeiro momento para depois seguirem sozinhos em busca dos seus objetivos. Os ciclistas se mantêm juntos, evitando a resistência do ar, até o ponto em que estão próximos da linha de chegada. Neste momento, como a cooperação não é mais interessante, cada ciclista gasta toda sua energia para seguir sozinho rumo à vitória.

Entre os animais, indivíduos irracionais, a cooperação também pode ser observada, inclusive entre animais de espécies diferentes [30]. Na migração das aves, é observado um exemplo de comportamento cooperativo entre animais de mesma espécie. Esses animais voam em conjunto, formando um desenho de seta, com o objetivo de diminuir a resistência do ar sobre seus corpos. As aves se revezam na posição que forma a ponta de uma seta, devido ao fato que neste ponto a força do ar incide com maior resistência. Dessa forma, as aves voam distâncias maiores com um gasto global menor de energia.

Atualmente, têm-se pesquisado e motivado o uso da cooperação entre máquinas.

No ambiente de uma rede sem fio, por exemplo, existem alguns cenários de cooperação. A comunicação entre os dispositivos integrantes de uma WLAN só é possível devido ao comportamento cooperativo realizado pelo ponto de acesso, que transmite os pacotes e gerencia o acesso ao meio compartilhado. Nesse caso, o ponto de acesso apresenta um comportamento cooperativo generoso, visto que gasta seus recursos cooperando com a rede sem esperar recompensa.

Nas redes *ad hoc*, a cooperação entre os dispositivos ocorre, principalmente, através do roteamento de pacotes. Devido à perda de uma infra-estrutura de comunicação, o encaminhamento cooperativo é essencial para a manutenção da conectividade e aumento da capacidade da rede [10]. Entretanto, na função de roteador, os nós sofrem um gasto nos seus recursos, causado pela transmissão dos pacotes de outros nós. Este consumo dos recursos pode limitar a participação dos nós no encaminhamento de pacotes, gerando um problema para a rede. O fato de permanecer com o rádio ligado durante a atividade cooperativa, também acarreta gastos de energia para os dispositivos. Portanto, nas redes *ad hoc*, os nós podem apresentar um comportamento cooperativo, porém, este comportamento deve ser motivado através de incentivos individuais ou de grupo.

Na Seção 2.3, foram descritos os tipos de redes *ad hoc* e suas aplicações. De acordo o tipo de rede *ad hoc*, existe ou não uma autoridade responsável por definir os objetivos da rede (ver Tabela 2.1). Nas redes definidas por uma autoridade, verifica-se a cooperação motivada por incentivos de grupo. Nessas redes, os nós são desenvolvidos para a execução de um objetivo comum que é definido pela autoridade. Entretanto, nas redes *ad hoc* civis, não há uma autoridade responsável pela definição dos objetivos da rede. Como, nesse tipo de rede, os dispositivos podem ter objetivos ou métricas de desempenho diferentes, a cooperação entre os nós não é natural, mesmo que isso acarrete na parcial ou completa desconexão da rede. Portanto, faz-se necessário o uso de incentivos individuais para motivar a cooperação entre os dispositivos. A Tabela 2.2 resume os tipos de motivação utilizados para cada tipo de rede *ad hoc* como pode ser visto em [5] [3] [31] [32].

Fitzek propõe em [32] uma classificação sobre os tipos de cooperação em ambientes de comunicação dividida em três classes: cooperação implícita, cooperação macro explícita e cooperação micro explícita. A principal diferença entre as classes é a necessidade ou não de utilizar benefícios como forma de estímulo ao comportamento cooperativo. Quando são utilizados incentivos, a motivação é explícita, por outro lado, quando não são utilizados incentivos, a motivação é implícita. A classe de cooperação explícita é dividida

em cooperação macro explícita e cooperação micro explícita.

**Tabela 2.2** Cooperação x Tipo de Rede *Ad Hoc*

| Rede | Autoridade | Incentivo |
|---|---|---|
| Militar | X | grupo |
| Sensoriamento | X | grupo |
| Resgate | X | grupo |
| Substituição Temporária | X | grupo |
| Civil | - | individual |

A cooperação implícita (passiva) é caracterizada pela ocorrência de justiça e respeito entre os indivíduos de forma passiva. Nessa classe, as entidades compartilham um dado recurso sem a necessidade de incentivos. Os protocolos de comunicação podem ser vistos como uma forma de cooperação implícita. Como exemplo de cooperação implícita, pode-se descrever o protocolo *Transmission Control Protocol* (TCP). Através desse protocolo de transporte, os usuários da *Internet* utilizam um mecanismo de controle de fluxo que previne o congestionamento dos roteadores. Portanto, os usuários utilizam somente uma parte da capacidade de enlace que é calculado pelo mecanismo de controle de fluxo.

A cooperação macro explícita é caracterizada pela utilização de incentivos como forma de estímulo ao comportamento cooperativo. O cenário de comunicação por múltiplos saltos, presente nas redes sem fio, é um exemplo desta classe de cooperação. O curto raio de alcance do sinal dos dispositivos é incrementado, virtualmente, através do encaminhamento por múltiplos saltos realizado pelos dispositivos intermediários da comunicação. Os indivíduos cooperativos são dispositivos finais sem fio, pontos de acesso virtuais ou roteadores sem fio. Este trabalho situa-se na classe de cooperação macro explícita, especificamente, sobre os dispositivos integrantes de uma rede *ad hoc*.

A cooperação micro explícita é semelhante à cooperação macro explícita, entretanto, os indivíduos que fazem parte desse grupo são unidades de processamento, partes funcionais do sistema e algoritmos. Um cenário representativo dessa classe de cooperação são as redes de quarta geração (4G Networks). Essa abordagem pretende integrar as redes de longo alcance (e.g., redes celulares) com as redes de curto alcance (e.g., WLAN e PAN). A possibilidade de cooperação é facilmente visualizada na comunicação *multicast* entre a estação base e um grupo de usuários. Como nas redes de curto alcance é verificado um consumo menor de energia do que o observado nas redes de longo alcance, os usuários

podem se revezar na comunicação com estação base e repassar as mensagens ao grupo, através da rede de curto alcance. Dessa forma, os serviços fornecidos pela rede celular também podem ser compartilhados entre os usuários através da rede de curto alcance.

### 2.6.1 Taxonomia do Comportamento

Os dispositivos móveis de uma de rede *ad hoc* civil podem apresentar vários tipos de comportamento, visto que não são gerenciados por uma entidade central ou subordinados (desenvolvidos) por uma autoridade. Portanto, são livres para buscar seus interesses, seja de forma cooperativa ou não.

Nas redes *ad hoc*, a participação cooperativa é mais representativa nas funções desempenhadas pela camada de rede [33] [5]. São funções dessa camada, a definição de rotas de transmissão e o encaminhamento de pacotes. Os nós que concordam em utilizar seus recursos na execução dessas funções são ditos cooperativos. Essa cooperação pode ser motivada através de incentivos de grupo ou individuais. Quando não é necessário o uso de incentivos à cooperação, tem-se o comportamento cooperativo generoso. Os nós que não cooperam com a rede, através da execução dessas funções, ou causam problemas relacionados à segurança, estão apresentando um comportamento indesejado.

O comportamento indesejado pode ser classificado em duas classes: comportamento malicioso e comportamento egoísta [34] [35] [3]. Os nós egoístas não possuem o objetivo de prejudicar outros nós, porém, quebram o paradigma da cooperação quando não cooperam, por exemplo, no roteamento de pacotes por múltiplos saltos. Os nós podem apresentar esse comportamento como resultado da tentativa de economizar seus recursos para futuras comunicações, ou pelo fato de cooperarem com a rede apenas quando estão utilizando seus serviços. Dessa forma, os nós egoístas causam problemas relacionados ao paradigma da cooperação.

Diferentemente dos nós egoístas, a prioridade dos nós maliciosos não é a economia de seus recursos. Os nós maliciosos são aqueles que utilizam a rede com o objetivo de prejudicar outros nós ou obter vantagens na utilização dos recursos da rede. São exemplos do comportamento malicioso a alteração das informações contidas nos pacotes de definição de rotas e a utilização de vários identificadores durante a permanência na rede (ataque *Sybil* [36]). Dessa forma, os nós maliciosos causam problemas relacionados à segurança da rede.

Os nós falhos ou posicionados em regiões desfavoráveis à atividade cooperativa (e.g., bordas da rede e regiões de gargalo) não devem ser confundidos com nós egoístas. As estratégias de incentivo à cooperação devem ser capazes de perdoar ou minimizar os efeitos punitivos sobre os nós que apresentam esse tipo de comportamento indesejado não intencional.

Conti et al. [5] definem o comportamento egoísta dividido em três classes: egoísta puro, egoísta avançado e o egoísta guloso. O comportamento egoísta ocorre quando o usuário sempre desliga o rádio transmissor durante o intervalo em que não utiliza os serviços da rede. Dessa forma, o usuário só coopera com a rede enquanto utiliza seus serviços. O comportamento egoísta avançado está relacionado ao não encaminhamento de pacotes através de modificações nos protocolos da camada de rede. Finalmente, o comportamento guloso ocorre quando benefícios são ofertados aos nós que compartilham seus recursos com a rede. Na tentativa de maximizar a quantidade de benefícios, os nós gulosos compartilham seus recursos em grande quantidade, provocando um desequilíbrio no compartilhamento de recursos da rede.

Buchegger e Le Boudee [37] não diferenciam o comportamento malicioso do comportamento egoísta, no entanto, neste trabalho a diferenciação é realizada visto que esses tipos de comportamento indesejado apresentarem conseqüências diferentes para as estratégias de incentivo à cooperação. A Figura 2.4 ilustra a taxonomia do comportamento em redes *ad hoc* de acordo com os comportamentos apresentados nesta Seção.

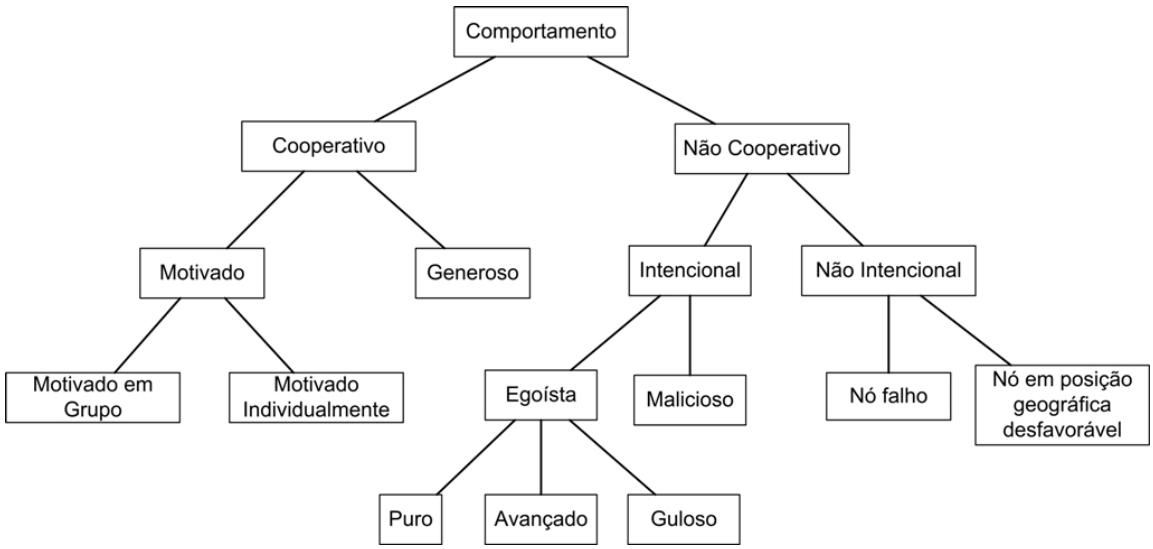



**Figura 2.4** Taxonomia do comportamento em redes *ad hoc*

### 2.6.2 Comportamento Indesejado nas Diferentes Camadas da Pilha de Protocolos

A necessidade de incentivar a cooperação ao nível das camadas da pilha de protocolos de comunicação é abordada por Obreiter e Klein [38], Conti et al. [5] e Fitzek e Katz [32]. Nesses trabalhos, os autores identificaram os possíveis tipos de comportamento indesejado que podem ser observados em cada uma das camadas.

Ao nível da camada MAC, os nós maliciosos podem modificar o mecanismo distribuído de controle de acesso ao meio compartilhado. Nas redes *ad hoc*, o mecanismo *Distributed Coordination Function* (DCF) é implementado pelo IEEE 802.11, com base no mecanismo *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) [14]. O mecanismo DCF determina que os nós devam respeitar um período aleatório de tempo (*backoff*) antes de acessar o meio para enviar ou reenviar os seus pacotes. O *backoff* é calculado aleatoriamente entre a faixa de valores [0, $CW$], em que $CW$ é janela de contenção (*Contention Window*) armazenada em cada nó. A cada ocorrência de colisão, um novo valor de *backoff* é calculado a partir do valor dobrado de $CW$. Esse processo é repetido até o pacote ser enviado com sucesso e o valor de $CW$ é dobrado até um valor máximo. Dessa forma, os nós maliciosos podem modificar o mecanismo DCF através da atribuição de valores mínimos para $CW$, visando à obtenção de vantagens no acesso ao meio compartilhado.

Ao nível da camada de rede, os nós egoístas não executam as funções de encaminhamento de pacotes de dados e/ou de controle (e.g., pacotes de definição de rotas), enquanto os nós maliciosos alteram as informações utilizadas pelo mecanismo de descoberta e manutenção de rotas. A estratégia proposta neste trabalho visa prevenir o comportamento egoísta no encaminhamento de pacotes.

Ao nível da camada de transporte, os nós maliciosos podem modificar o mecanismo de controle de congestionamento do protocolo TCP. Esse protocolo é originário das redes cabeadas e limita a capacidade do enlance para prevenir ou diminuir o congestionamento sobre os roteadores da *Internet*. Portanto, os nós cooperativos concordam em utilizar esse mecanismo para melhorar o desempenho global da rede. Com o objetivo de enviar seus fluxos utilizando a capacidade máxima do enlace, os nós maliciosos burlam o protocolo TCP através de modificações nesse mecanismo. Os nós maliciosos também atacam a rede através do não encaminhamento das mensagens de confirmação de entrega (pacotes do tipo *acknowledgment* - ACK) ou do envio pela rede de mensagens de erro sobre as rotas, o que afeta o desempenho do TCP.

Os protocolos da camada de aplicação são responsáveis pela disponibilização dos serviços da rede. O compartilhamento de serviços tem sido bastante pesquisado em redes P2P e alguns de seus problemas são herdados pelas redes *ad hoc* [39]. Os nós egoístas compartilham poucos ou nenhum serviço, enquanto os nós maliciosos disponibilizam serviços falsos ou tentam burlar o sistema de prestação prioritária de serviços aos nós reconhecidamente cooperativos. A Tabela 2.3 resume os tipos de comportamentos indesejado em cada uma das camadas da pilha de protocolos de comunicação.

**Tabela 2.3** Comportamento indesejado sobre as camadas

| **Camada** | **Comportamento Indesejado** | |
|---|---|---|
| | **Egoísta** | **Malicioso** |
| MAC | - | cálculo do *backoff* |
| Rede | Descarte de pacotes (dados e controle) | Alteração das informações de roteamento |
| Transporte | Descarte do ACK | Burlar controle de fluxo<br>Aviso falso de falhas |
| Aplicação | Não compartilhamento de recursos | Compartilhamento falso de recursos<br>Uso de identidade falsas |

### 2.6.3 Incentivando a Cooperação em Redes Ad Hoc

Nos últimos anos, tem-se observado o desenvolvimento de diversos trabalhos que abordam o tema cooperação em redes *ad hoc*. A maioria desses trabalhos defende a necessidade de incentivar a cooperação entre os dispositivos móveis, entretanto, Felegyhazi [40] e Buttyan [23] abordam a possibilidade de manter a conectividade da rede sem a necessidade de incentivar a cooperação através de estímulos externos. A dependência mútua existente entre os nós da rede seria suficiente para provocar um equilíbrio cooperativo sobre o comportamento dos nós. Dessa forma, pelo menos uma estratégia cooperativa é seguida pelos nós, mesmo que não motivada.

Os mecanismos que estimulam a cooperação através de incentivos defendem que o uso de estímulos externos é indispensável para a manutenção da conectividade da rede. Esses mecanismos tentam tornar todos ou alguns tipos de comportamento indesejado (ver Seção 2.6.1) não atrativos aos nós da rede. A heterogeneidade dos dispositivos representa um grande desafio para esse tipo de estratégia, visto que a capacidade cooperativa dos

elementos pode variar de acordo com a capacidade de seus recursos.

A Figura 2.5 ilustra os tipos de incentivo à cooperação em redes *ad hoc* abordados com maior freqüência na literatura. Inicialmente, foram desenvolvidos os mecanismos de detecção e punição do comportamento indesejado através do uso de reputação. Em seguida, foram abordados trabalhos de incentivo à cooperação baseados em comércio virtual. Recentemente, têm-se desenvolvido mecanismos mais dinâmicos, baseados em teoria dos jogos que utilizam as duas técnicas anteriores.

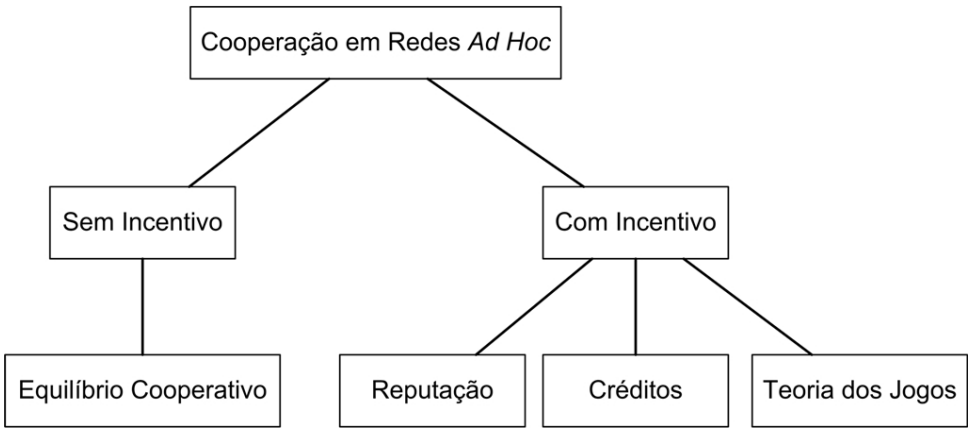



**Figura 2.5** Cooperação em redes *ad hoc*

### 2.6.4 Baseados em Reputação

Nos mecanismos baseados em reputação, os nós avaliam o comportamento de seus vizinhos através do monitoramento das transmissões (modo promíscuo habilitado). Dessa forma, os nós podem analisar se algum tipo de comportamento indesejado, como o não encaminhamento de pacotes (egoísta) ou a mudança nas informações de roteamento (malicioso), está sendo executado pelos seus vizinhos. Caso um nó apresente um comportamento indesejado repetidas vezes, ele é taxado como falho ou indesejado, o que representa uma baixa reputação na rede. Esses mecanismos executam as funções de monitoração, construção da reputação e resposta ao comportamento indesejado. Como punição ao comportamento indesejado, os nós não cooperativos são isolados do encaminhamento de pacotes e enfrentam dificuldade na tentativa de estabelecer suas comunicações. Portanto, a cooperação é incentivada através de punições sobre o comportamento indesejado.

Marti et al. [41] propuseram uma ferramenta de monitoração do comportamento, chamada *watchdog*. Quando um nó apresenta um comportamento indesejado, sua re-

putação é divulgada em toda a rede (informação de segunda mão). Os autores criaram outra ferramenta chamada *pathrater* que evita o uso de nós egoístas na formação das rotas. O trabalho em conjunto dessas duas ferramentas procura isolar os nós egoístas do encaminhamento de pacotes, no entanto, para isso, inundam a rede com informações de reputação.

Buchegger e Le Boudec [42] apresentam um mecanismo semelhante, chamado *CONFIDANT*, que também identifica e isola os nós não cooperativos. Entretanto, esse mecanismo dificulta o encaminhamento dos pacotes dos nós egoístas como forma de punição. A construção da reputação é realizada através de inferência bayesiana sobre o comportamento dos nós com o objetivo determinar a ocorrência de comportamento indesejado. Esse mecanismo utiliza o conceito de confiança para permitir que os nós compartilhem informações locais de reputação com o restante da rede.

O mecanismo *CORE* [34], desenvolvido por Michiardi e Molva, utiliza além das informações negativas (reputação) as informações positivas sobre o comportamento dos nós. O uso de informações positivas (comportamento cooperativo) tem o objetivo de dificultar a alteração maliciosa sobre as informações de segunda mão. Com o objetivo de não prejudicar os nós falhos ou os nós situados em posições desfavoráveis à atividade cooperativa (encaminhamento de pacotes), é permitida a reintegração dos nós marcados como não cooperativos, caso estes comecem a cooperar. Refaei et al. [43] propõem uma abordagem semelhante, no entanto, os nós determinam o comportamento de seus vizinhos sem a necessidade de compartilhar informações.

He et al. [44] definem o mecanismo *SORI* em que a reputação de um determinado nó é calculada através da probabilidade com que ele encaminha os pacotes de seus vizinhos. Quando um nó recebe um pacote de um determinado vizinho, ele o encaminha com a mesma probabilidade com que este vizinho encaminha os seus pacotes. Essa abordagem não necessitam de uma entidade central e as informações de reputação de um nó são propagadas apenas entre os seus vizinhos.

### 2.6.5 Baseados em Créditos

Os mecanismos baseados em créditos incentivam a cooperação entre os nós através do uso de dinheiro virtual. Neste tipo de abordagem, os nós que encaminham pacotes recebem créditos pela execução desse serviço, enquanto que os nós que originam pacotes perdem

créditos no encaminhamento de seus pacotes. Os nós são motivados a cooperarem com a rede pelo fato de só conseguirem se comunicar caso possuam créditos suficientes para o encaminhamento de seus pacotes. Dessa forma, a cooperação é a única forma de obter os créditos necessários para o estabelecimento das comunicações.

Buttyán e Hubaux apresentam o projeto *Terminode* [45], que é considerado o primeiro a utilizar o conceito de comércio virtual como estímulo ao comportamento cooperativo. Os autores propõem duas estratégias baseadas no uso de dinheiro virtual (*nuglet*) que é utilizada pelos nós como um meio de pagamento pelos serviços prestados. A primeira estratégia, *Packet Purse Model* (PPM), determina que cada nó deva pagar pelo serviço de repasse de pacotes a todos os nós intermediários que participam do encaminhamento dos seus pacotes. Dessa forma, o pacote deve ser preenchido com *nuglets* suficientes para pagar pelo serviço de encaminhamento a todos os nós intermediários envolvidos nessa transmissão. Para garantir que cada nó intermediário retire apenas a quantidade de *nuglets* do pacote que lhe é devida, é proposto o uso de um módulo de segurança implementado em *hardware*.

A segunda estratégia, *Packet Trade Model* (PTM), determina que o pacote não deva carregar os *nuglets* na sua estrutura. Nesse caso, o pacote é comercializado entre os nós intermediários. Cada nó intermediário compra o pacote do nó anterior e vende para o próximo nó sucessivamente até que nó destino seja alcançado. Essa segunda estratégia apresenta a desvantagem de repassar a despesa ao nó destino. Ataques baseados em negação de serviço (*Denial of Service* - DoS) podem ser utilizados para zerar a quantidade de créditos de um determinado nó.

Buttyán e Hubaux [46] apresentam uma continuação do projeto *Terminode* em que foram realizadas algumas modificações na estratégia PPM. Os nós gerenciam seus créditos e a quantidade de carga da bateria restante. Foi apresentado um modelo matemático relativamente simples, que tem como principal objetivo maximizar o envio de pacotes enquanto minimiza a taxa de descarte. Embora os autores tenham melhorado a abordagem, o funcionamento correto da estratégia continua refém da existência de um módulo de segurança correto e honesto que gerencie o sistema de créditos.

Zhong et al. desenvolveram o mecanismo *SPRITE* [47] que também utiliza o conceito de dinheiro virtual abordado pelo projeto *Terminode*, entretanto, substitui o uso de um módulo de segurança em cada nó por uma entidade central na qual todos confiam (*Credit Clearance Service* - CCS). Esse mecanismo implementa um sistema de

recibos que contabiliza a quantidade de pacotes originados e encaminhados por cada nó. Os nós devem mandar seus recibos para a entidade CCS, que é responsável por administrar o sistema de créditos e definir a quantidade de créditos a ser paga pelo serviço de encaminhamento de pacotes. A principal desvantagem desse mecanismo é o uso de uma entidade central de administração que representa um ponto único de falha para o mecanismo.

Raghavan e Snoeren [48] propõem que os nós cooperativos tenham encaminhamento prioritário de seus fluxos. Para os nós com poucos créditos é fornecido o encaminhamento por melhor esforço. Dessa forma, os nós que ainda não estão interessados no encaminhamento prioritário ou estão em uma posição geográfica desfavorável para atividade cooperativa possam estabelecer comunicações. Esse mecanismo determina que cada nó deva armazenar as rotas para todos os destinos possíveis e que seu comportamento seja conhecido por todos os nós da rede. Dessa forma, cada encaminhamento de pacote deve ser informado a todos os nós da rede, o que provoca uma inundação da rede pelos pacotes de controle.

A estratégia de incentivo à cooperação proposta neste trabalho utilizada o conceito de dinheiro virtual para possibilitar que o sistema de encaminhamento prioritário de pacotes seja capaz beneficiar os nós cooperativos.

### 2.6.6 Teoria dos Jogos

Teoria dos jogos é o campo da matemática aplicada que descreve e analisa situações de tomada de decisão [49]. Ferramentas de predição de resultados são desenvolvidas a partir desse campo para analisar interações complexas entre entidades racionais.

As estratégias baseadas em teoria dos jogos utilizam modelos matemáticos chamados de jogos. Os jogos são formados por três componentes: jogadores, estratégias (ações) e as funções de utilidade. Esses componentes modelados em um ambiente de rede *ad hoc* representam, respectivamente, os dispositivos móveis, as ações que um jogador pode tomar (e.g., encaminhar ou não um pacote, qual mecanismo de economia de energia utilizar, seleção do esquema de modulação) e as métricas de desempenho dos jogadores (e.g., *throughput*, retardo, descarte, consumo de energia).

O resultado do jogo é analisado por cada jogador através de sua função de utilidade que representa seus objetivos. Geralmente, os resultados são representados por

*payoffs* (benefícios) atribuídos a cada jogador de acordo com o resultado de suas ações sobre os demais. A melhor ação de um jogador é aquela que maximiza sua função de utilidade para um dado conjunto de ações executadas pelos outros jogadores. Dessa forma, cada jogador possui um conjunto de ações (estratégias) que podem ser seguidas com o objetivo de maximizar seus *payoffs*.

Quando, em um jogo, nenhum jogador consegue obter um maior valor de *payoff* através da mudança unilateral de sua estratégia, tem-se o Equilíbrio de Nash (EN) [23]. As estratégias que abordam a existência de cooperação sem a necessidade de incentivos (ver Seção 2.6.3) analisam as situações em que a rede converge para um EN cooperativo. Entretanto, não há garantias de que esse resultado seja o melhor possível, ou o desejado pelos jogadores. Dessa forma, tem-se utilizado um conceito da economia chamado de ótimo de Pareto que determina se o resultado alcançado é o melhor possível [49]. Um resultado é ótimo de Pareto se o valor da função de utilidade de um jogador não pode ser melhorado sem que haja perda na função de utilidade de um outro jogador.

Atualmente, esses conceitos têm sido utilizados para analisar a eficiência dos mecanismos baseados em incentivos. Para isso, as características de uma rede *ad hoc* são modeladas para um ambiente de jogo. A teoria dos jogos também está sendo utilizada no desenvolvimento de estratégias mais dinâmicas de incentivo à cooperação, em que são analisadas as interações entre os dispositivos e o custo-benefício de compartilhar serviços em detrimento do consumo de recursos.

Urpi et al. [31] apresentam um modelo matemático baseado em teoria dos jogos em que várias características de uma rede *ad hoc* são modeladas (e.g., mobilidade, comportamento egoísta, encaminhamento de pacotes, energia). O cálculo do *payoff*, que é atribuído a cada ação dos nós, é realizado em função da capacidade cooperativa de cada nó. Dessa forma, o tratamento da heterogeneidade dos nós proporciona uma maior robustez ao modelo, visto que o comportamento de um nó é interpretado de acordo com a capacidade real de seus recursos. Nesse trabalho, os autores utilizaram o modelo proposto para analisar as principais estratégias de incentivo à cooperação.

Srinivasan et al.[33] definem um modelo em que os nós são divididos em classes de acordo com seus níveis de energia. Os autores analisaram o custo-benefício do uso de energia *versus* os *payoffs* adquiridos. É apresentado um modelo matemático para encontrar o ótimo de Pareto de cada jogador sobre suas restrições de energia. Uma das estratégias analisadas é a *Tit-for-Tat* (TFT) [29] em que a decisão de um nó encaminhar

ou não um pacote é tomada de acordo com as experiências passadas com os outros nós. Dessa forma, o comportamento atual de um nó é resultado das interações com os outros nós da rede. Essas experiências podem ser vistas como uma espécie de reputação par-a-par entre os nós da rede.

## 2.7 COMPARAÇÃO ENTRE OS MECANISMOS DE INCENTIVO À COOPERAÇÃO

As abordagens apresentadas neste Capítulo atuam no domínio da camada de rede e possuem o objetivo de incentivar a cooperação no encaminhamento de pacotes. Embora o objetivo seja o mesmo, as estratégias utilizadas possuem características específicas que apresentam vantagens e desvantagens.

A Tabela 2.4 apresenta as características suportadas pelas principais estratégias de incentivo à cooperação apresentadas neste Capítulo. O combate ao comportamento egoísta e malicioso são apresentadas por todas as estratégias com exceção da estratégia *SPRITE* que é vulnerável ataques, visto que é requerida a comunicação com a entidade central para atualizar a quantidade de créditos. O uso de informações compartilhadas é característico das estratégias baseadas em reputação, pelo fato de difundirem a informação de reputação, enquanto o uso de um sistema de segurança é característico das estratégias baseadas em créditos, em que o comercio precisa ser protegido. O uso de administração central pode ser visto nas estratégias baseadas em créditos e nas estratégias baseadas em reputação, no entanto, o uso do modo promíscuo é uma característica das estratégias baseadas em reputação, visto que necessitam da monitoração para analisar o comportamento dos nós vizinhos.

As estratégias baseadas em reputação são capazes de detectar e punir os comportamentos egoísta e malicioso, no entanto, com exceção da estratégia *CORE* [34], não são capazes de reabilitar um nó que passou a apresentar um comportamento cooperativo. Outra desvantagem dessa abordagem é que seu melhor desempenho está diretamente ligado à inundação da rede com pacotes de controle e ao uso de uma entidade central para gerenciar as informações de comportamento.

A principal vantagem das estratégias baseadas em créditos é a ausência do compartilhamento de informações e da monitoração de comportamento. Dessa forma, evita-se que um nó malicioso manipule as infamações de reputação. Essa vantagem é assegura pela necessidade dos nós possuírem créditos para transmitirem os seus pacotes, o que é

verdade e extensível aos nós maliciosos e egoístas. No entanto, para gerenciar o comércio virtual é necessário o uso de um módulo de segurança ou de uma entidade central de administração. Nesse tipo de abordagem, quanto menor for a mobilidade da rede maior será a dificuldade dos nós de borda se comunicarem, visto que não terão chance de encaminhar pacotes como forma de adquirir créditos.

As principais vantagens das abordagens baseadas em teoria dos jogos são o dinamismo e a robustez, visto que as características da rede *ad hoc* são modeladas matematicamente para prover um apoio á tomada de decisão no momento de encaminhar um pacote. Entretanto, a modelagem matemática da rede requer uma maior complexidade de desenvolvimento.

**Tabela 2.4** Mecanismos de Incentivo à Cooperação

| **Características Suportadas** | **Mecanismos de incentivo à Cooperação** (Ação na Camada de Rede) | | | | | |
|---|---|---|---|---|---|---|
| | *Watchdog* | *CONFIDANT* | *SORI* | *CORE* | *Terminode* | *SPRITE* |
| Combate Egoísta | X | X | X | X | X | X |
| Combate Malicioso | X | X | X | X | X | - |
| Permite Reabilitação | - | - | - | X | X | X |
| Informações Locais | - | - | X | - | X | - |
| Informações Compartilhadas | X | X | X | X | - | - |
| Administração Central | - | X | - | X | - | X |
| Sistema de Segurança | - | - | - | - | X | X |
| Modo Promíscuo | X | X | X | X | - | - |
| Reputação | X | X | X | X | - | - |
| Créditos | - | - | - | - | X | X |
| Teoria dos Jogos | - | - | - | - | - | X |

# CAPÍTULO 3

# ESTRATÉGIA DE INCENTIVO À COOPERAÇÃO

Neste Capítulo, é descrita a estratégia de incentivo à cooperação baseada em créditos em que o comportamento cooperativo é motivado pelo provisionamento de QoS. Inicialmente, a Seção 3.1 aborda o funcionamento básico da estratégia de incentivo à cooperação. A Seção 3.2 descreve as suposições que serviram de base para o desenvolvimento da estratégia proposta e caracterização da rede *ad hoc* alvo. Na Seção 3.3 é detalhado o Sistema de Créditos em que são abordados os tipos de informações utilizadas e como os créditos são contabilizados e manipulados. A Seção 3.4 descreve o mecanismo de encaminhamento prioritário que escalona os pacotes de acordo com a quantidade de créditos dos nós origem. Por fim, a Seção 3.5 analisa a portabilidade da estratégia quando utilizados protocolos de roteamento que não fornecem rota completa de envio dos pacotes.

## 3.1 VISÃO GERAL

A estratégia proposta neste trabalho visa o incentivo à cooperação em redes *ad hoc*. Seu funcionamento básico está voltado ao encaminhamento prioritário de pacotes dos nós cooperativos. Para isso, foi desenvolvido um sistema de créditos que é utilizado como parâmetro pelo mecanismo de encaminhamento prioritário. O mecanismo de contabilização e manutenção de créditos utiliza apenas informações locais, que estão disponíveis na camada de rede (e.g., protocolo de roteamento), para atualizar os dados que são utilizados pelo sistema de créditos. Aos nós com baixa quantidade de créditos (e.g., créditos negativos) é fornecido um serviço com qualidade inferior.

Neste trabalho, os próprios nós são responsáveis pelo gerenciamento dos créditos, visto que não são utilizadas entidades centrais de gerenciamento. A ausência de uma administração central foi possível devido à presença de dois fatores chave:

- **O escopo das informações utilizadas é local:** o sistema utiliza apenas informações de primeira mão (experiência local), dessa forma, não há necessidade de compartilhar informações sobre os créditos.

- **O gerenciamento de créditos é descentralizado:** a quantidade de créditos de um nó é gerenciada de forma local e independente dos outros nós da rede, dessa forma, o mecanismo de encaminhamento prioritário não requisita qualquer tipo de informação ao nó que está sendo servido.

## 3.2 SUPOSIÇÕES

O desenvolvimento do mecanismo de incentivo à cooperação proposto foi realizado com base em um conjunto de suposições sobre as características da rede *ad hoc*. Essas suposições, que são apresentadas e justificadas a seguir, são utilizadas na modelagem e desenvolvimento do mecanismo proposto e contextualizam para quais tipos de cenários ele é indicado.

- **Rede *ad hoc* civil:** nas redes gerenciadas por uma autoridade comum (ver Seção 2.3) o comportamento cooperativo é natural, o que dispensa uma estratégia de incentivo à cooperação. Embora o mecanismo também possa ser utilizado em outros tipos de redes *ad hoc*, a busca por uma boa QoS nas comunicações pode ser visualizada com maior facilidade nas redes *ad hoc* civis.
- **Necessidade de comunicação:** os nós integrantes da rede têm o desejo de estabelecer comunicações com outros nós. O mecanismo proposto prioriza os fluxos dos nós cooperativos, dessa forma, os nós interessados em comunicações com boa QoS devem cooperar com a rede para obter vantagens em suas comunicações.
- **Antenas omnidirecionais e simétricas:** todos os nós possuem antenas de transmissão omnidirecionais simétricas com a mesma potência de transmissão.
- **Modo promíscuo habilitado:** o mecanismo proposto utiliza informações contidas no cabeçalho dos pacotes. Essas informações são importantes tanto para os nós que fazem parte da rota como para os nós que estão na vizinhança da rota. Através do modo promíscuo, cada nó recebe os pacotes transmitidos pelos seus vizinhos, mesmo quando esses pacotes não são endereçados a ele.
- **Conhecimento completo da rota:** a rota que o pacote deve seguir entre os nós origem e destino, é conhecida por todos os nós intermediários envolvidos nessa comunicação. Neste trabalho, é utilizado o protocolo de roteamento DSR [9] que atende essa suposição.

- **Presença do comportamento egoísta:** os nós integrantes da rede podem apresentar um comportamento egoísta no encaminhamento dos pacotes de dados e de definição de rotas. Esta estratégia está voltada à prevenção do comportamento egoísta sobre as funções da camada de rede (ver Seção 2.6.2).
- **Ausência do comportamento malicioso:** não é foco deste trabalho o combate ao comportamento malicioso. Dessa forma, os nós podem causar problemas relacionados ao paradigma da cooperação, no entanto, não podem causar problemas relacionados à segurança da rede (ver Seção 2.6.2).
- **Identificador único:** cada nó possui um identificador único que não é alterado.

## 3.3 SISTEMA DE CRÉDITOS

Neste trabalho, é utilizado como estímulo à cooperação o provisionamento de encaminhamento prioritário aos fluxos dos nós cooperativos. Para isso, foi desenvolvido um sistema de créditos que contabiliza tanto o trabalho cooperativo dos nós intermediários como o uso do serviço de encaminhamento pelos nós origem.

A contabilização dos créditos é semelhante à apresentada pelo projeto *TERMINODE* [45], no entanto, os créditos não são armazenados e manipulados dentro dos pacotes, mas sim, na Tabela de Créditos Local (TCL). Dessa forma, evita-se a necessidade de um módulo de segurança para gerenciar a manutenção dos créditos. Outra diferença é que a TCL não armazena informações sobre a quantidade de créditos do nó local, visto que essa informação não é compartilhada.

Diferentemente da abordagem *SPRITE* [47], a estratégia proposta não utiliza uma entidade central de administração de créditos, portanto, os créditos são administrados pelos próprios nós de forma local e descentralizada. Dessa forma, eliminam-se os problemas inerentes ao uso de um ponto único de falha e a necessidade do uso de comunicação segura com este ponto.

### 3.3.1 Informações Utilizadas

Cada nó da rede possui uma TCL que é utilizada pelo sistema de créditos para gerenciar e manter a quantidade de dinheiro virtual dos nós. Nessa tabela são armazenadas as

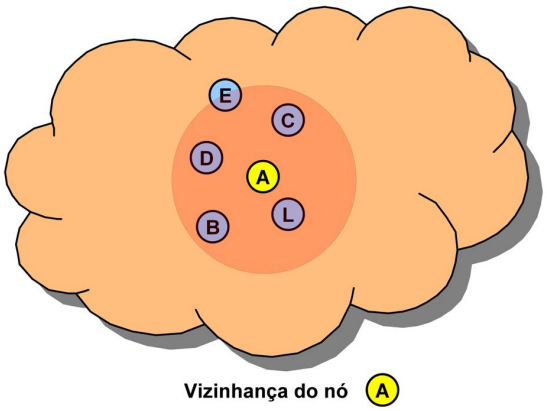
seguintes informações: o identificador e a quantidade de créditos dos nós conhecidos.

O termo *nós conhecidos* está relacionado à porção da rede com que cada nó já cooperou (encaminhou pacotes), utilizou os serviços (teve seus pacotes encaminhados) ou verificou o encaminhamento cooperativo através do uso do modo promíscuo.

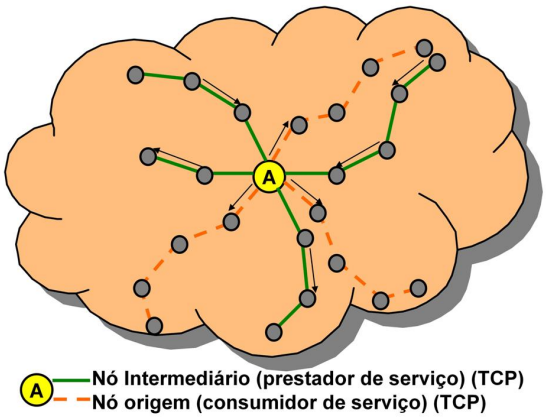
A Figura 3.1 ilustra os cenários em que um nó é considerado conhecido por outro. Inicialmente, a Figura 3.1(a) ilustra o cenário mais simples em que um nó é considerado conhecido. Nessa Figura o nó $A$ considera como conhecido todos os nós que estão dentro do seu raio de transmissão.

Quando o nó é originador ou intermediário de uma rota, ele conhece todos ou parte dos nós dessa rota, de acordo com o protocolo de transporte utilizado. Quando é fornecida a confirmação de entrega, como no caso do TCP, é possível ter certeza que todos os nós intermediários cooperaram na entrega do pacote até o destino. Entretanto, quando utilizado o UDP, essa confirmação não é fornecida. Nesse caso, quando o nó é intermediário da rota ele considera como conhecido apenas o nó origem, os nós intermediários antecessores na rota e o próximo salto, que é conhecido pelo fato do nó escutar a transmissão de seus vizinhos. Quando o nó é origem da rota, ele considera como conhecido apenas o primeiro salto da rota pelo fato de escutar apenas essa transmissão. A Figura 3.1(b) ilustra os nós considerados conhecidos pelo nó $A$ quando este é originador ou intermediário de rotas transportadas pelo TCP, enquanto a Figura 3.1(c) ilustra o mesmo cenário em que é utilizado o protocolo de transporte UDP. Quando o nó é o destino da rota, ele considera como conhecido todos os nós envolvidos na comunicação, independente do protocolo de transporte utilizado.

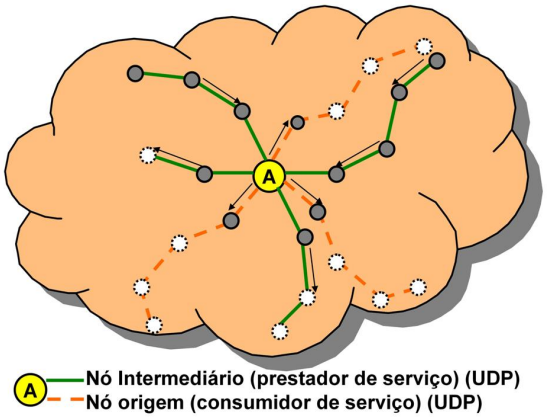
Os uso do modo promíscuo e da transmissão omnidirecional permitem que os nós analisem o cabeçalho dos pacotes enviados ou encaminhados pelos seus vizinhos. Dessa forma, ao analisar o cabeçalho do pacote encaminhado por um de seus vizinhos, o nó considera como conhecido o nó que originou o pacote e os nós intermediários, novamente, de acordo com o protocolo de transporte utilizado. No caso do TCP, o nó conhece a rota completa, enquanto no caso do UDP, o nó conhece apenas a parte da rota entre o nó originador do pacote e o seu vizinho que está encaminhado o pacote. A Figura 3.1(d) ilustra o encaminhamento de pacotes pelo nó $B$, que está na vizinhança do nó $A$, quando utilizado o protocolo de transporte TCP. Já a Figura 3.1(e) ilustra o mesmo cenário com uso do protocolo de transporte UDP. Neste caso, apenas parte da rota é considerada conhecida.



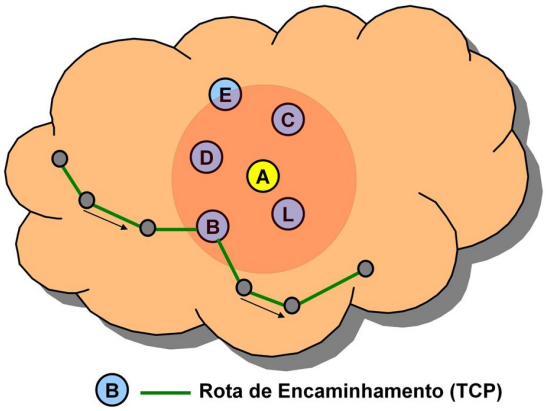

(a) Vizinhança



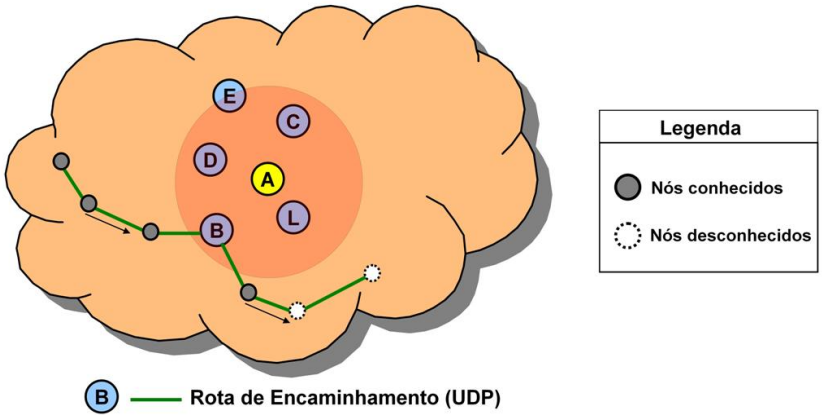

(b) Parte da rota - TCP




(c) Parte da rota - UDP




(d) Modo promíscuo - TCP




(e) Modo promíscuo - UDP

**Figura 3.1** Nós conhecidos

A rota do pacote também é utilizada pelo sistema de créditos, no entanto, esse tipo de informação não é mantido ou atualizado pela TCL, devido à sua volatilidade e pelo fato que essa informação pode ser acessada no cabeçalho dos pacotes.

### 3.3.2 Contabilização dos Créditos

Os créditos de um nó conhecido são atualizados na TCL de acordo com seu comportamento corrente. A quantidade de créditos do nó que encaminha pacotes é incrementada a cada pacote encaminhado, enquanto que a quantidade de créditos do nó que origina pacotes é decrementada a cada pacote enviado. A quantidade inicial de créditos dos nós é igual à zero. Dessa forma, para manter uma quantidade de créditos positiva, os nós precisam encaminhar pacotes.

A quantidade atual de créditos de um nó conhecido $i$ é calculada pela seguinte equação:

$$C_i = \beta Pf_i - \alpha Pe_i, \quad \beta > \alpha, \tag{3.1}$$

em que $\beta$ é a quantidade de créditos atribuída a cada pacote encaminhado pelo nó conhecido $i$, $\alpha$ é a quantidade de créditos debitada a cada pacote enviado pelo nó $i$, $Pf_i$ é a quantidade total de pacotes encaminhados pelo nó $i$, e $Pe_i$ é a quantidade total de pacotes enviados pelo nó $i$. O valor da equação 3.1 é utilizado pelo mecanismo de encaminhamento prioritário como parâmetro de prioridade no encaminhamento de pacotes.

As estratégias baseadas em créditos encontradas na literatura (ver Seção 2.6.5) não trabalham com créditos negativos. Quando a quantidade de créditos de um nó chega à zero, este nó só volta a conseguir estabelecer conexões quando adquirir créditos. Entretanto, a estratégia proposta neste trabalho, permite que um nó sem créditos ou com créditos negativos tenha seus pacotes encaminhados. Dessa forma, um nó que apresenta um tipo de comportamento egoísta não intencional (ver Seção 2.6.1) pode estabelecer comunicações, no entanto, essas comunicações sofrem com descartes e retardos maiores do que os oferecidos aos nós cooperativos (QoS baixa). Portanto, o mecanismo de encaminhamento prioritário, que é explicado em detalhes na Seção 3.4, trabalha também com valores negativos relativos à equação 3.1.

A quantidade de créditos $C_i$ de um nó $i$ pode assumir valores diferentes na TCL dos nós que possuem uma entrada para nó $i$. Como as informações de créditos não são compartilhadas entre os nós e a relação de consumo (envio de pacotes) e prestação de

serviços (encaminhamento de pacotes) pode variar muito entre os pares, dificilmente a quantidades de créditos $C_i$ é a mesma em todas as TCLs dos nós que possuem uma entrada para o nó $i$.

### 3.3.3 Manutenção da TCL

O sistema de créditos atualiza as informações contidas na TCL de forma local e descentralizada. Os dois tipos de operações possíveis sobre a TCL são a criação de novas entradas e a atualização da quantidade de créditos sobre as entradas existentes. O processo de manutenção utiliza apenas as informações contidas no cabeçalho dos pacotes. Através dessas informações são identificados o nó fonte, os nós intermediários e o nó destino. Neste trabalho, uma entrada nunca é apagada, dessa forma, evita-se que um nó egoísta seja beneficiado pelo reinício da sua quantidade de créditos (quantidade inicial de créditos de uma nova entrada é igual à zero).

Novas entradas são adicionadas à TCL de um nó em diversas situações. A Figura 3.1 ilustra os cenários em que o nó $A$ pode criar novas entradas na sua TCL. A Figura 3.1(a) ilustra o cenário em que são criadas entradas referentes à vizinhança do nó $A$. Na Figura 3.1(b) o nó $A$ pode criar novas entradas para todos os nós que fazem parte das rotas de pacotes enviados ou encaminhados pelo nó $A$, visto que é utilizado o TCP como transporte. Nesse caso é criada uma entrada para o nó destino, visto que ele coopera no envio do ACK. A Figura 3.1(d) ilustra o cenário semelhante em que também é utilizado o transporte TCP, no entanto, o nó $A$ não faz parte da rota. Quando utilizado o UDP, como nas Figuras 3.1(c) e 3.1(e), são criadas entradas para os nós originadores e para a parte dos nós intermediários conhecidos pelo nó $A$.

O processo de atualização da TCL ocorre quando um nó recebe um pacote. O resultado dessa execução pode ter resultados diferentes de acordo com as seguintes variáveis:

- **O nó faz parte da rota:** o nó tem acesso às informações do pacote pelo fato de ser origem, intermediário ou destino da rota.
- **O nó não faz parte da rota:** através do uso do modo promíscuo, os nós podem analisar os pacotes que trafegam no seu raio de transmissão.
- **O pacote é do tipo dados:** para os pacotes desse tipo, o DSR armazena todos os saltos entre a origem e o destino.

- **O pacote é do tipo definição de rotas (DSR):** os três tipos de pacote do DSR são o RREQ, o RREP e o RRER.
- **O protocolo de transporte é o TCP:** como esse protocolo oferece confirmação de entrega é possível ter certeza que os nós intermediários cooperaram no encaminhamento dos pacotes.
- **O protocolo de transporte é o UDP**: não há confirmação de entrega, dessa forma, apenas o encaminhamento dos nós antecessores ao nó corrente e do próximo salto pode ser computado.

#### 3.3.3.1 Pacotes de Dados

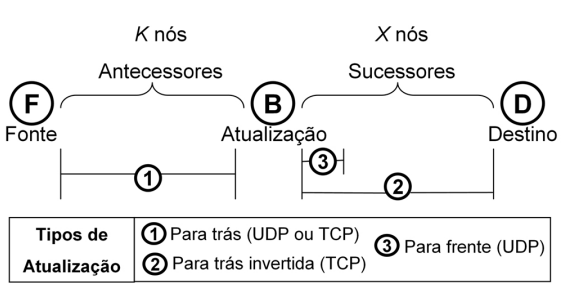
Inicialmente é descrito o processo de atualização da TCL quando recebido um pacote de dados. Esse processo ocorre tanto no caso em que o nó faz parte da rota como no caso em que o nó está na vizinhança da rota.

No primeiro caso (ver Figura 3.2(a)), o nó decrementa a quantidade de créditos do nó fonte, que requisitou o serviço, e incrementa a quantidade de créditos dos $k$ nós intermediários, que estão provendo o serviço, entre ele e o nó fonte (atualização para trás). Os créditos dos $x$ nós intermediários entre ele e o nó destino são atualizados, em um momento futuro, se o protocolo de transporte fornecer confirmação de entrega. Caso não haja essa confirmação, são atualizados apenas os créditos do sucessor imediato na rota, quando o nó corrente escutar a transmissão desse salto (atualização para frente).

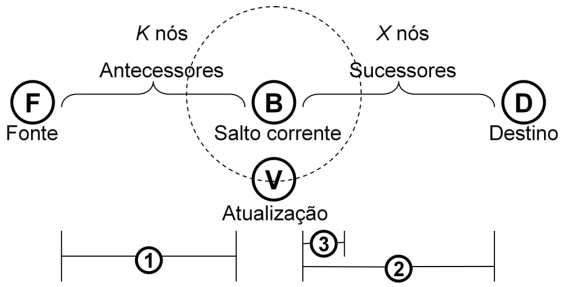
Quando o protocolo de transporte fornece confirmação de recebimento, cada nó que recebe essa confirmação executa a atualização para trás invertida (nó destino como origem do pacote ACK) utilizando a rota do pacote ACK como forma de identificar os nós intermediários sucessores. Nesse caso, o trabalho cooperativo dos nós intermediários é mantido com maior justiça, visto que todos os integrantes da rota executam essa atualização. O encaminhamento do ACK é motivado pelo fato dos nós intermediários sucessores terem seu trabalho cooperativo computado apenas quando esse pacote é retornado.

No segundo caso (ver Figura 3.2(b)), em que o nó não faz parte da rota, ele decrementa a quantidade de créditos do nó origem e incrementa a quantidade de créditos dos $k$ nós intermediários entre o nó vizinho e o nó fonte. Os créditos do nó vizinho também são incrementados. A atualização dos créditos dos $x$ nós intermediários sucessores ao seu vizinho é realizada de forma semelhante ao primeiro caso, no entanto, quando não houver

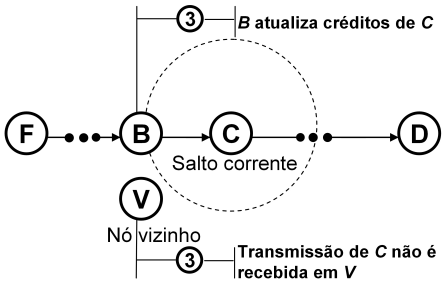
confirmação, a transmissão do sucessor imediato ao seu vizinho pode está fora do seu raio de alcance, o que impossibilita a atualização para frente (ver Figura 3.2(c)).




(a) Atualização pelo nó que faz parte da rota




(b) Atualização pelo nó que é vizinho da rota




(c) Atualização para frente - executada apenas em $B$

**Figura 3.2** Tipos de atualização - pacotes de dados

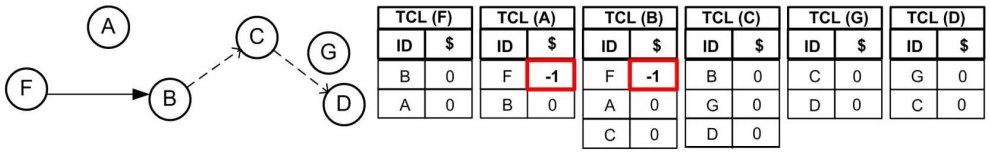
Em ambos os casos, a quantidade de créditos do nó destino é mantida apenas quando utilizado o transporte TCP, visto que o envio do pacote ACK pelo nó destino aumenta a justiça na contabilização dos créditos. Em [45] é descrita a abordagem PTM em que o nó destino é debitado pelo envio de pacotes do nó origem. Entretanto, os autores constataram que essa característica poderia ser utilizada pelos nós maliciosos para decrementar a quantidade de créditos dos nós destinos (ver Seção 2.6.5). Neste trabalho, a quantidade de créditos do nó destino é incrementada quando este coopera com os processos de confirmação de entrega e descoberta de rotas.

A Figura 3.3 ilustra o processo de atualização de cada TCL envolvida no envio de um pacote entre o nó fonte $F$ e o nó destino $D$. Nesse exemplo, é assumido que o protocolo de transporte não fornece confirmação de entrega e que a rota já foi determinada, no entanto, para facilitar o entendimento, o processo de manutenção da TCL executado na definição de rotas é desconsiderado, dessa forma, os nós estão, inicialmente, com a quantidade de créditos igual a zero e possuem entradas apenas para seus vizinhos. As setas tracejadas indicam os saltos que compõem a rota e a seta cheia indica o salto corrente

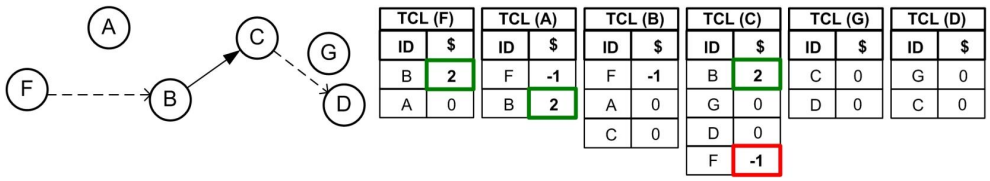
da transmissão. Nesse exemplo, os valores de $\alpha$ e $\beta$ da equação 3.1 são, respectivamente, 1 e 2.

A Figura 3.3(a) ilustra o resultado do processo de atualização após o envio do pacote do nó $F$ para o nó $B$. A quantidade de créditos do nó originador $F$ é decrementada na TCL dos nós $B$ e $A$. O nó $B$ teve acesso ao pacote por fazer parte da rota, enquanto o nó $A$ analisou o pacote através do uso do modo promíscuo. A quantidade de créditos do nó intermediário $B$ não é incrementada na tabela de seus vizinhos pelo fato de não ser possível, nesse momento, ter certeza que o comportamento cooperativo foi executado.

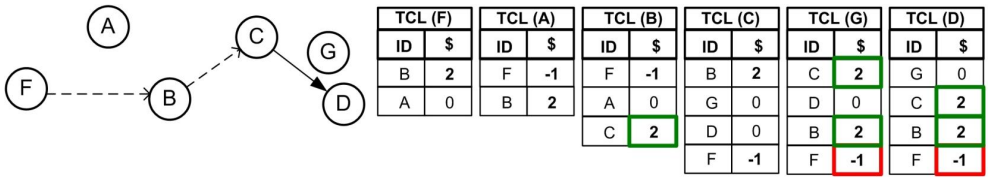
A Figura 3.3(b) ilustra o resultado do processo de atualização após o encaminhamento do pacote do nó $B$ para o nó $C$. Nesse caso, ocorrem as atualizações para trás e para frente. Através do uso do modo promíscuo os nós $F$ e $A$ executam a atualização para frente em que a quantidade créditos do próximo salto (nó $B$) é incrementada. O nó $C$ executa a atualização para trás em que a quantidade de créditos do nó fonte (nó $F$) é decrementada e a quantidade de créditos dos nós intermediários antecessores (nó $B$) é incrementada.




(a) Após envio do pacote de $F$ para $B$




(b) Após envio do pacote de $B$ para $C$




(c) Após envio do pacote de $C$ para $D$

**Figura 3.3** Atualização da TCL - transporte UDP

Por fim, a Figura 3.3(c) ilustra o resultado do processo de atualização após o encaminhamento do pacote do nó intermediário $C$ para o nó destino $D$. Nesse caso,

também ocorre as atualizações para frente e para trás. O nó $B$ incrementa a quantidade de créditos do nó $C$, através da atualização para frente, enquanto os nós $G$ e $D$ decrementam a quantidade de créditos do nó $F$ e incrementam a quantidade de créditos dos nós intermediários $B$ e $C$ através da atualização para trás.

**Algoritmo 1**: Atualização da TCL - Pacotes de dados

s **Entrada**: Pacote $P_{dados}$
**Resultado**: Tabela local atualizada
**Dados**: $IP_{Local}$, $IP_{Fonte}$, $Rota_{pacote}[]$, $TCL[]$

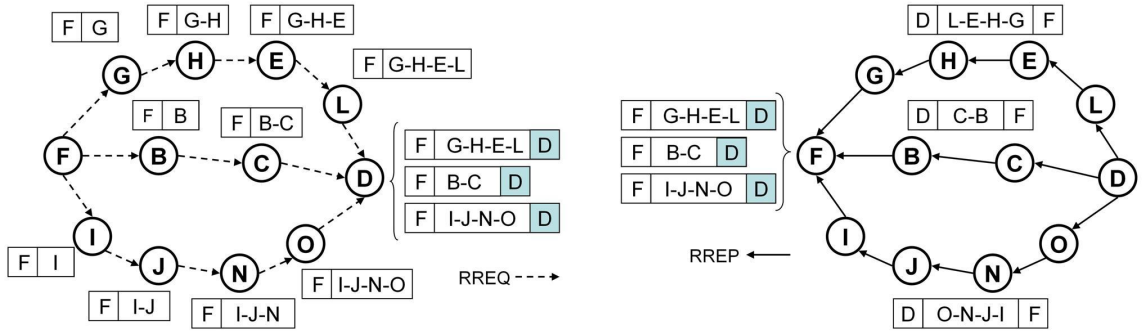
```
1  Os dados são inicializados com as informações do pacote aqui
2  para i ← 1 até Comprimento_rota faça
3      se IP_Fonte == Rota_pacote[i] então
4          TCL[Rota_pacote[i]] ← TCL[Rota_pacote[i]] - α          // Para trás
5      fim
6      se IP_Local == Rota_pacote[i] então
7          se TemConfirmacao () então
8              AtualizaAposConfirmacao (P)          // Para trás invertida
9              retorna
10         fim
11         senão
12             AtualizaAposTransmissao (P)                // Para frente
13             retorna
14         fim
15     fim
16     TCL[Rota_pacote[i]] ← TCL[Rota_pacote[i]] + β              // Para trás
17 fim para
```

O processo de atualização da TCL, em que o nó corrente faz parte da rota e que o pacote analisado é do tipo dados, é descrito no Algoritmo 1. Esse algoritmo recebe como entrada o pacote em que são obtidas as informações necessárias para atualizar os créditos dos nós envolvidos nessa comunicação. Na linha 1, os dados são inicializados com as informações do cabeçalho do pacote. O vetor $Rota_{pacote}[]$ contém a seqüência dos identificadores dos nós envolvidos na comunicação, desde a origem até o destino. A TCL é indexada através dos identificadores dos nós, dessa forma, a quantidade de créditos de um nó $i$ da rota é obtida através de $TCL[Rota_{pacote}[i]]$. As linhas 3 a 5 atualizam a quantidade de créditos do nó fonte, enquanto a linha 16 atualiza a quantidade de créditos dos nós intermediários antecessores ao nó corrente (nó destino ou intermediário). Nas linhas 5 a 15 é verificado se todos os nós antecessores já foram creditados e qual estratégia utilizar para creditar o restante dos nós. Quando utilizado o transporte TCP, o restante dos nós in-

termediários são creditados pela função *AtualizaAposConfirmacao(P)*, enquanto quando utilizado UDP, o próximo salto é creditado através da função *AtualizaAposTransmissao*.

**3.3.3.2 Pacotes de Definição de Rotas** O processo de atualização da TCL é executado também no envio e encaminhamento dos pacotes de definição de rotas. Neste trabalho, é assumido que o protocolo de roteamento utilizado pela rede *ad hoc* é o DSR. Esse protocolo é abordado na Seção 2.4.1 em que são descritos os mecanismos de descoberta e manutenção de rotas.

O mecanismo de descoberta de rotas utiliza dois tipos de pacotes: o RREQ, que é enviado por difusão até alcançar o nó destino, e o RREP, que é enviado ao nó origem contendo a rota solicitada. O pacote RREQ mantém um registro em que são armazenados os identificadores dos nós visitados até alcançar o nó destino (ver Figura 3.4(a)). O RREP armazena a rota completa entre a origem e o destino que é resultante do processo de difusão do RREQ (ver Figura 3.4(b)). O pacote RRER é utilizado pelo mecanismo de manutenção de rotas para comunicar ao nó origem a existência de uma falha em um dos saltos da rota. O nó intermediário que identifica a falha envia o pacote RRER ao nó origem. Esse pacote armazenada a rota que deve ser seguida para alcançar o nó destino. Dessa forma, as informações contidas nos registros desses pacotes podem ser utilizadas de maneira semelhante às informações contidas nos pacotes de dados.




(a) Envio do pacote RREQ (b) Envio do pacote RREP

**Figura 3.4** Informações sobre a rota dos pacotes DSR

O processo de atualização da TCL ocorre de maneira diferente para cada um desses tipos de pacotes. No caso do pacote RREQ, o processo de atualização é semelhante ao executado no recebimento de um pacote de dados sobre o transporte UDP. Como o registro do RREQ possui apenas os nós já visitados, são executadas apenas as atualizações

para trás e para frente. A atualização para trás invertida não é executada pelo fato de ser enviada confirmação de entrega para pacotes do tipo DSR.

No caso do pacote RREP, o envio utiliza a seqüência inversa dos nós visitados pelo RREQ (ver Seção 2.4.1), dessa forma, o nó destino do processo de descoberta é atribuído como nó origem no envio desse pacote. Entretanto, o processo de atualização não interpreta dessa forma, visto que o solicitante do processo de descoberta de rotas é o nó origem. Portanto, o processo de atualização é executado de forma semelhante à atualização para trás invertida, visto que o envio do RREP é resultado da cooperação dos nós na difusão do pacote RREQ. Os créditos do nó origem são decrementados, enquanto os créditos do nó destino são incrementados, visto que este cooperou com o processo de descoberta de rotas.

Quando são utilizadas antenas e enlaces direcionais, o processo de envio do RREP ocorre através de um novo processo de descoberta de rotas na direção contrária (ver Seção 2.4.1). Nesse caso, a atualização ocorre de forma semelhante à efetuada sobre os pacotes RREQ, no entanto, a quantidade de créditos do nó origem, que solicitou o primeiro processo de descoberta de rotas, é decrementada.

O processo de atualização da TCL só é executado quando os pacotes estão trafegando na vizinhança de alcance do sinal. Uma alternativa seria atualizar todas as TCLs a cada pacote enviado na rede como foi proposto em [48], no entanto, essa alternativa acarreta uma grande sobrecarga dos recursos da rede em que o aumento da justiça, na contabilização dos créditos, não promove, necessariamente, um aumento no desempenho da rede.

## 3.4 ENCAMINHAMENTO PRIORITÁRIO

De acordo com o nível do tráfego e densidade da rede, um nó pode rotear diversos fluxos ao mesmo tempo [50] [51]. Neste trabalho, os pacotes de encaminhamento são armazenados em fila e são servidos de acordo com a quantidade de créditos dos nós origem. O mecanismo de encaminhamento prioritário é ilustrado na Figura 3.5. O mecanismo é composto de três componentes: classificador, filas e escalonador.

O componente classificador determina em qual fila cada pacote de encaminhamento deve ser armazenado. Nessa fase, a quantidade de créditos fornecida pela equação 3.1 é utilizada na classificação dos pacotes. Caso a quantidade de créditos do nó origem seja

positiva, o pacote é armazenado em uma das filas de maior prioridade, enquanto se a quantidade de créditos do nó origem for negativa, o pacote é armazenado em uma das filas de menor prioridade. Dessa forma, a classificação é feita inicialmente entre duas classes: *Positiva* e *Negativa*. A classe *Positiva* é dividida nas subclasses *Alta* e *Média*, enquanto a classe *Negativa* é dividida nas subclasses *Normal* e *Baixa*. O processo de classificação se repete na escolha da subclasse.

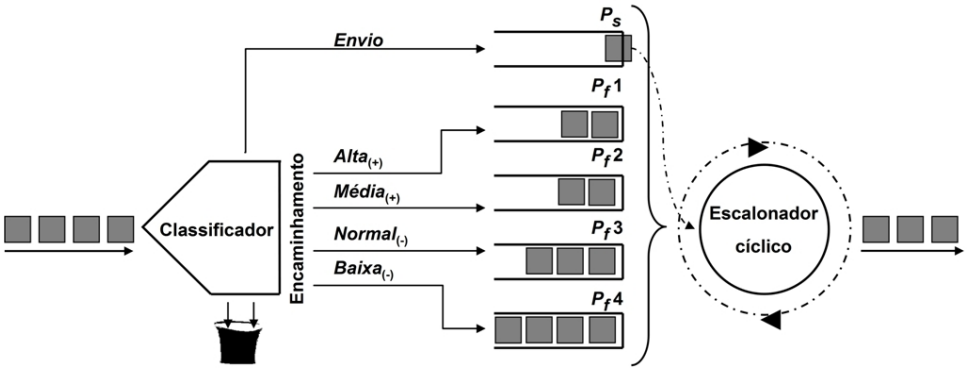



**Figura 3.5** Fila dos pacotes de encaminhamento

Este mecanismo utiliza um número de quatro filas que é referente ao número de subclasses. Cada fila possui uma prioridade de serviço diferente que é chamada de peso. Os pesos são definidos de acordo com a classe da fila, dessa forma, o peso $P_f1$ está relacionado à subclasse *Alta*, enquanto o peso $P_f4$ está relacionado à subclasse *Baixa*. Quanto maior a prioridade da subclasse, maior é o peso da fila relacionada ($P_f1 > P_f2 > P_f3 > P_f4$). As filas possuem a mesma capacidade de armazenamento e são do tipo *First In First Out* (FIFO) em que a ordem dos pacotes armazenados na fila não é alterada.

O componente escalonador foi desenvolvido a partir da disciplina de escalonamento WRR. O WRR organiza os pacotes em classes e os escalona de acordo com os pesos dessas classes. A cada rodada, esse escalonador cíclico serve uma determinada quantidade de pacotes, de cada classe, que é definida pelos pesos. Dessa forma, evita-se o problema de *starvation* que é verificado quando o escalonamento basea-se apenas em prioridades [28].

No caso de baixo nível de energia, o nó pode optar por descartar os pacotes de baixa prioridade e manter o encaminhamento apenas para os nós com alta prioridade (e.g., subclasses *Alta* e *Média*). Esse descarte funciona como uma espécie de penalidade sobre os nós com créditos negativos. Nesse caso, o comportamento egoísta não intencional não é perdoado, no entanto, o nó que está com baixa quantidade de energia pode continuar

encaminhando pacotes dos nós cooperativos.

Sempre que um nó tenta transmitir um pacote, ele compete pelo meio compartilhado, independentemente do pacote ser de encaminhamento ou originado pelo próprio nó. Dessa forma, existe uma classe para os pacotes que são originados pelo próprio nó. Esses pacotes disputam o escalonamento com os pacotes de encaminhamento e são armazenados na fila referente à classe *envio*. Essa fila possui um peso $P_s$ de serviço.

O componente escalonador cíclico permite o desenvolvimento de um mecanismo de controle do grau do comportamento cooperativo dos nós. Dessa forma, os nós podem ser mais ou menos cooperativos de acordo com os valores dos pesos das filas de encaminhamento e envio. Nós que precisam de créditos podem aumentar, durante um intervalo de tempo, a prioridade das filas de encaminhamento, enquanto os nós com muitos créditos pode priorizar o envio de seus pacotes. Através do uso de políticas, esse mecanismo pode controlar o grau cooperativo do nó baseado em uma série de variáveis relacionadas as suas métricas de desempenho (grau de cooperação x benefícios desejados). O gerenciamento dinâmico do comportamento cooperativo é uma investigação futura deste trabalho.

## 3.5 COMENTÁRIOS

O sistema de gerenciamento de créditos e o mecanismo de encaminhamento prioritário, desenvolvidos neste trabalho, foram construídos ao nível da camada de rede com base no protocolo de roteamento DSR. Entretanto, esses mecanismos não são dependentes ao uso desse protocolo.

Quando o protocolo de roteamento é baseado em tabela de rotas, em que as entradas armazenam apenas o endereço do salto anterior e do próximo salto, a estratégia proposta possui um menor alcance de gerenciamento. Nesse caso, são contabilizados apenas os créditos dos nós origem e destino (quando for o caso), e dos nós antecessor e sucessor diretos. O endereço dos nós destino e origem é obtido através do cabeçalho IP, enquanto o endereço dos nós antecessor e sucessor são obtidos através da tabela de rotas. A contabilização executada através do modo promíscuo não sofre alterações, visto que essa característica é inerente ao protocolo utilizado pela camada MAC (IEEE 802.11). Portanto, mesmo quando utilizado outro tipo de protocolo de roteamento, a estratégia proposta é capaz de gerenciar os créditos dos nós, no entanto, o alcance do gerenciamento é menor devido à limitação das informações de rota.

Embora a estratégia proposta rodando sobre o TCP resulte em maior justiça na contabilização dos créditos, o uso de várias filas de encaminhamento aumenta a chance de inversões na ordem de chegada dos pacotes ao destino, o que reduz o desempenho desse protocolo. Outra desvantagem do TCP, dessa vez inerente ao seu uso em MANETs, está relacionada com as limitações do seu controle de congestionamento nesse tipo de cenário. Essa limitações são abordadas na Seção 4.2 do próximo Capítulo que descreve os testes de configuração e desempenho.

# CAPÍTULO 4

# IMPLEMENTAÇÃO E AVALIAÇÃO

Neste capítulo são descritos a implementação da estratégia proposta e os experimentos. A Seção 4.1 descreve, em alto nível, as adaptações que implementam a estratégia sobre o DSR do *Network Simulator*, versão 2.30 (NS-2) [52]. A Seção 4.2 descreve a avaliação da estratégia através da técnica de simulação. Inicialmente, são descritos os experimentos de calibragem da estratégia. Em seguida, são descritos o testes comparativos realizados sobre duas implementações do NS-2: uma original, que foi chamada de *DSR puro*; e outra, modificada, que foi chamada de *DSR + estratégia*.

## 4.1 IMPLEMENTAÇÃO

Este trabalho foi desenvolvido ao nível da camada de rede em que são executadas as funções de definição de rotas e encaminhamento de pacotes. Atualmente, o NS-2 disponibiliza a implementação de quatro protocolos de roteamento para MANETs (e.g., DSDV, DSR, AODV e TORA) [53]. Este trabalho foi implementado através de adaptações ao protocolo de roteamento DSR.

### 4.1.1 DSR implementado pelo NS-2

O protocolo de roteamento DSR foi desenvolvido no NS-2, em parte, pelo projeto *CMU Monarch* [54]. Essa implementação disponibiliza as principais funcionalidades do DSR que foram propostas por Johnson e Maltz em [9].

O DSR é implementado através de vários arquivos (classes), no entanto, nem todos são utilizados. Os principais arquivos são: o $dsragent.[cc, h]$ que implementa o agente de roteamento, o $hdr\text{-}src.[cc, h]$ que constrói o cabeçalho de pacotes utilizado pelo DSR, o $path.[cc, h]$ que armazena os identificadores dos nós da rota, o $srpacket.h$ que representa o pacote do tipo DSR e encapsula o cabeçalho *hdr-src* e o $dsr\text{-}priqueue.[cc, h]$ que

implementa a fila de prioridades do DSR. A implementação desse trabalho foi realizada sobre os arquivos *dsragent.*$[cc, h]$ e *dsr-priqueue.*$[cc, h]$.

O arquivo *dsragent.*$[cc, h]$ implementa três formas distintas em que um pacote pode ser recebido pelo agente de roteamento (camada de rede): normal, em que o nó é destino ou intermediário do pacote recebido, promíscuo, em que o nó está na vizinhança de transmissão do pacote e por motivo de falha, em que o pacote é retornado pela camada MAC. As três formas de entrada de pacotes são implementadas, respectivamente, pelas funções *recv()*, *tap()* e *xmitFailed()*.

A Figura 4.1 ilustra o diagrama de estados[4] do DSR após o recebimento normal de um pacote (função recv()). Inicialmente, o cabeçalho *source route* do pacote é extraído para ser verificado se está bem formado. O cabeçalho está bem formado quando possui uma rota completa de envio ou quando o pacote do tipo de definição de rotas (RREQ ou RREP). Caso o cabeçalho esteja bem formado, o endereço IP de destino do pacote indica qual é o próximo estado (estados modificados). Quando o IP destino é igual ao IP corrente, o pacote chegou ao seu destino (estado *HandlePacketReceipt*). Quando o IP destino é diferente do IP corrente, o pacote deve ser encaminhado. O tipo do pacote determina qual estratégia de encaminhamento deve ser utilizada (estado *HandleForwarding* - pacote de dados - ou *HandleRouteRequest* - pacote RREQ). Cada um desses estados são implementados através de funções que recebem os respectivos nomes.

A função *tap()* é utilizada pelo agente DSR para otimizar o processo de descoberta de rotas. Os nós são capazes de, promíscuamente, analisar os pacotes que trafegam na sua vizinhança. Dessa forma, os nós têm acesso às rotas em que os pacotes estão trafegando, mesmo quando não participam da comunicação. Como o armazenamento das rotas observadas enriquece o repositório de rotas, um número menor de processos de descoberta de rotas é exigido.

A função *xmitFailed()* é executada quando a camada MAC não consegue enviar o pacote. Nesse caso, é executado o processo de manutenção de rotas em que um pacote RRER é enviado em direção ao nó origem para comunicá-lo da existência de falha em um dos saltos da rota.

[4]Adaptado do sítio $< http://www.winlab.rutgers.edu/\ zhibinwu/html/DSR_ns2.html >$

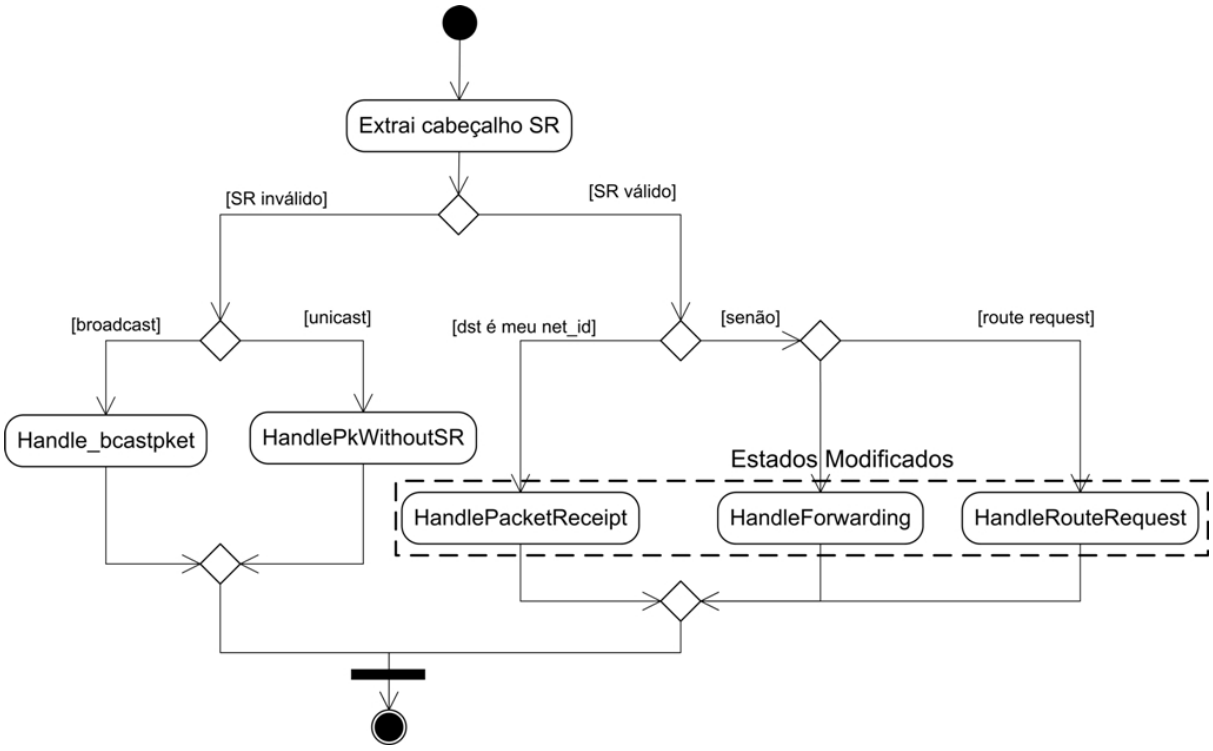



**Figura 4.1** Diagrama de estado - função *recv()*

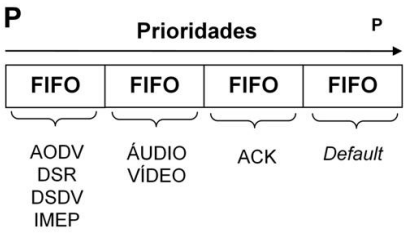
### 4.1.2 Adaptações ao DSR/NS-2

No Capítulo 3 foram descritos o sistema de créditos e o mecanismo de encaminhamento prioritário que compõem a estratégia de incentivo à cooperação proposta neste trabalho. Cada uma das funcionalidades apresentadas no Capítulo 3 foi desenvolvida como uma extensão à implementação do DSR e sua fila de pacotes. Essas funcionalidades são descritas a seguir:

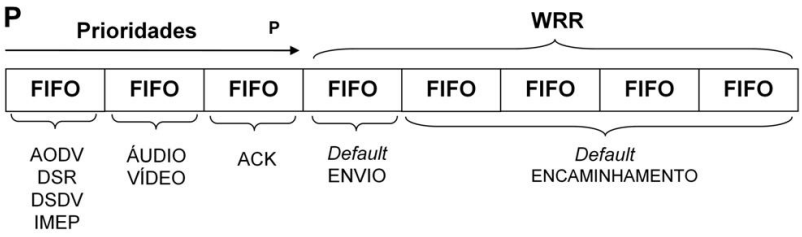
- **Contabilização dos créditos:** a contabilização dos créditos dos nós envolvidos na transmissão de um pacote é executada pelas funções que implementam a criação de entradas e atualização dos créditos (e.g., para frente, para trás e para trás invertida). O Algoritmo1 (ver Seção 3.3.3.1) apresenta a implementação dos tipos de atualização da TCL quando o pacote recebido é do tipo de dados. Para os pacotes de definição de rotas, o algoritmo é semelhante, no entanto, a forma como os créditos do nó origem e do destino são atualizados varia de acordo com o tipo desse pacote (ver Seção 3.3.3.2). Cada uma das três funções utilizadas pelo agente DSR para receber um pacote foi adaptada para executar os processos de manutenção da

TCL.

- **Nova fila de pacotes:** o agente DSR utiliza quatro filas do tipo FIFO para o armazenamento de pacotes de acordo com seus tipos, como mostrado na Figura 4.2(a). As filas são escalonadas através do uso de prioridades. Uma fila de menor prioridade é servida quando todas as filas de maior prioridade não possuem mais pacotes. Neste trabalho, foi mantido o uso das três primeiras filas, no entanto, a quarta fila, que armazena os pacotes de envio e encaminhamento, foi dividida em uma fila de envio e quatro filas de encaminhamento, como mostrado na Figura 4.2(b). As filas de envio e encaminhamento são escalonadas por uma adaptação à disciplina WRR como descrito na Seção 3.4.




(a) Fila DSR




(b) Fila DSR modificada pela estratégia

**Figura 4.2** Fila de pacotes do DSR

- **Comportamento egoísta:** o DSR foi desenvolvido com base na hipótese de que os nós são cooperativos [9], no entanto, em uma rede civil não há garantias de que essa hipótese se torne verdade. Neste trabalho, um nó egoísta aceita participar das rotas, no entanto, descarta todos os pacotes de dados que devem ser encaminhados. Esse comportamento, foi desenvolvido com base na descrição dos testes efetuados pelos trabalhos baseados em reputação (e.g., *Watchdog-Pathrater* [41], CORE [34] e CONFIDANT [42]). Dessa forma, um nó sempre é cooperativo na definição de rotas, no entanto, pode ser egoísta no encaminhamento de pacotes. O nó egoísta é implementado dessa forma para ter mais chance de descartar pacotes, visto que

no caso em que ele não participa do processo de definição de rotas, também não participa do encaminhamento de pacotes. Esse comportamento foi implementado, inclusive, para a versão original do NS-2, visto que para realizar a comparação entre o *DSR puro* e o *DSR + estratégia* os cenários devem ser idênticos.

- **Comportamento cooperativo com graduação:** a configuração das filas de envio e encaminhamento determina a razão de pacotes enviados e encaminhados a cada ciclo do escalonador. Dessa forma, um nó que prioriza os pacotes de envio aos pacotes de encaminhamento pode ser considerado menos cooperativo do que um nó que serve as filas de envio e encaminhamento na mesma proporção. O efeito da variação do grau de comportamento cooperativo sobre os fluxos roteados também é analisado através de simulações.

## 4.2 TESTES

Foram realizados dois tipos de testes: calibragem e avaliação. O objetivo dos testes de calibragem é a configuração da estratégia através de um cenário simples e controlado. No teste de avaliação, o comportamento da estratégia, configurada, foi analisado através de cenários mais complexos em que foram utilizados um maior número de nós e mobilidade. A implementação do *DSR puro* foi utilizada como base de comparação para a estratégia proposta.

Os nós se comunicam através de um tráfego *Constante Bit Rate* (CBR) sobre o transporte UDP. No nível de enlace, foi utilizado o IEEE 802.11b (DCF, 2Mbps). Os nós possuem antenas omnidirecionais simétricas com a mesma potência de transmissão e um raio de alcance de 250 metros.

Os testes não foram realizados sobre o protocolo de transporte TCP devido aos problemas impostos pelas redes *ad hoc* sobre o seu controle de congestionamento. Como o desenvolvimento do TCP foi voltado para as redes infra-estruturadas, as diversas formas de perda de pacotes (e.g., colisões, nós ocultos, quebra de rotas, falhas de *hardware*, particionamento da rede, dentre outros) observadas nas MANETs são tratadas como problemas de congestionamento, o que reduz consideravelmente o desempenho do TCP nesse cenário [55]. Dessa forma, o uso do TCP pode mascarar a análise dos testes sobre a estratégia proposta.

Foram avaliadas as seguintes métricas: Percentil 90 do retardo fim-a-fim, taxa de

descarte e Percentil 90 do *jitter*. O retardo fim-a-fim é calculado pelo tempo total em que cada pacote leva para chegar ao seu destino. A taxa de descarte é obtida pela razão entre o número de pacotes que alcançaram um determinado destino sobre o número total de pacotes que foram originados pela respectiva fonte. O *jitter* representa a variação do retardo fim-a-fim e é calculado através dos resultados de retardos sucessivos.

O tempo de cada simulação e o número de rodadas utilizadas foram calculados com base nas técnicas: critério de parada e remoção do estado transiente. Essas técnicas são apresentadas por Jain em [56]. A remoção do estado transiente determina a partir de qual instante a simulação alcança um estado consistente, em que pode ser realizado algum tipo de análise. Essa técnica influencia na determinação do tempo de duração de cada simulação. O critério de parada determina o número de replicações independentes (sementes de geração de números aleatórios diferentes) da simulação que são necessárias para gerar bons resultados em relação ao nível de confiança desejado. Dessa forma, o nível de confiança desejado sobre os resultados a serem analisados é alcançado através do cálculo do tempo de simulação e do número de replicações independentes (rodadas). As replicações são independentes, visto que a cada rodada é utilizada uma semente de geração de números aleatórios inédita.

### 4.2.1 Testes de Calibragem

Os testes de calibragem foram realizados com o objetivo de configurar os parâmetros que influenciam o comportamento da estratégia. Ao final desse processo, são escolhidos, através de uma análise empírica, os valores em que a estratégia alcançou os melhores resultados.

- **Sistema de créditos:** variáveis $\alpha$ e $\beta$ da equação de contabilização dos créditos. A variável $\alpha$ determina o valor do débito, enquanto a variável $\beta$ determina o valor do crédito. Esses valores determinam o quão cooperativo um nó deve ser para permanecer com créditos positivos. O valor de $\beta$ foi definido como um múltiplo do valor de $\alpha$, em que $\beta > \alpha$.
- **Mecanismo de encaminhamento prioritário:** peso da fila de envio ($P_s$), pesos das filas de encaminhamento ($P_f1, P_f2, P_f3, P_f4$) e o tamanho dessas filas em número máximo de pacotes ($T_f$). Os pesos determinam o grau de cooperação que um nó exerce sobre o encaminhamento dos pacotes.

Nesta primeira fase de testes, foi utilizado um cenário simples com o objetivo de executar uma simulação controlada. Dessa forma, o resultado das simulações é influenciado basicamente pelas variáveis em análise.

**Tabela 4.1** Configuração da simulação - testes de calibragem

(a) Parâmetros da simulação

| Parâmetro | Valor |
|---|---|
| Cenário (área) | 700m x 500m |
| Alcance do Sinal | 250m |
| Movimento | Estático |
| MAC | IEEE 802.11b (DCF, 2Mbps) |
| Antena | Omnidirecional e Simétrica |
| Aplicação | CBR |
| Transporte | UDP |
| Tamanho do Pacote | 512 B |
| Taxa (pcts/s) | 5, 10, 15, 20, 25, 30, 35 |
| Tempo de Simulação | 900 s |
| Número de rodadas | 10 |

(b) Parâmetros da estratégia - primeira etapa

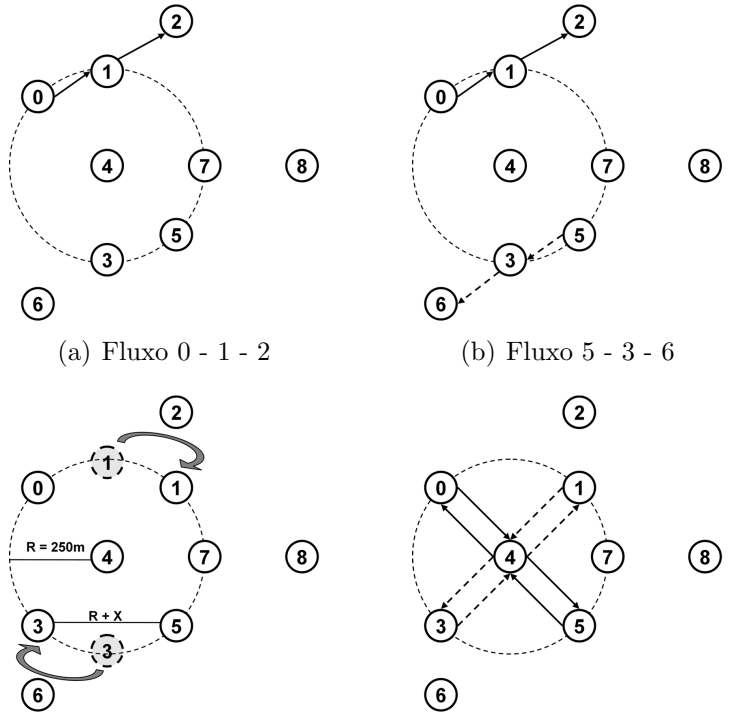
| Parâmetro | Valor |
|---|---|
| $\beta$ | (2), 1 e 3 |
| $\alpha$ | (1) |
| $P_s$ | - |
| $P_f1$ | (8), 4, 5 e 6 |
| $P_f2$ | (4), 3 e 4 |
| $P_f3$ | (2) |
| $P_f4$ | (1) |
| $T_f$ | (50), 10, e 30 (Pacotes) |

(c) Parâmetros da estratégia - segunda etapa

| Parâmetro | Valor |
|---|---|
| $\beta$ | 2 |
| $\alpha$ | 1 |
| $P_s$ | 4, 6, 8 e 10 |
| $P_f1$ | 8 |
| $P_f2$ | 4 |
| $P_f3$ | 2 |
| $P_f4$ | 1 |
| $T_f$ | 50 (Pacotes) |

Os testes são realizados em duas etapas. Na primeira etapa, é testada a variação dos seguintes valores: $\beta$, $T_f$, $P_f1$ , $P_f2$, $P_f3$, e $P_f4$. O resultado dessa etapa é a definição das variáveis que controlam o mecanismo de encaminhamento prioritário. Na segunda etapa, é testada a variação de $P_s$, mantendo-se fixos os valores definidos na primeira etapa. O resultado dessa etapa é a análise do grau do comportamento cooperativo em que um nó pode priorizar ou não o envio de seus pacotes.

A Tabela 4.1 apresenta a configuração da simulação e da estratégia. A Tabela 4.1(b) apresenta os possíveis valores de cada parâmetro analisado na primeira etapa e suas configurações base (valores entre parênteses). Para os testes de um parâmetro, enquanto sua variação é analisada, os demais parâmetros permanecem com o respectivo valor base. A Tabela 4.1(c) apresenta a configuração da segunda etapa em que é variado apenas o valor de $P_s$.




(a) Fluxo 0 - 1 - 2

(b) Fluxo 5 - 3 - 6

(c) Disposição em quadrado

(d) Encaminhamento sem envio

(e) Encaminhamento com envio

**Figura 4.3** Cenário utilizado nos testes de calibragem

O cenário utilizado nas duas etapas é constituído de nove nós como ilustrado na Figura 4.3. Inicialmente, o nó 1 apresenta um comportamento cooperativo no encami-

nhamento de pacotes do nó 0 ao nó 2 (ver Figura4.3(a)). Em seguida, o nó 3 coopera no encaminhamento de pacotes do nó 5 ao nó 6 (ver Figura 4.3(b)). Nesse primeiro momento, tem-se dois nós adquirindo créditos ($C_1 > C_3 > 0$) e dois nós perdendo créditos ($C_0 < C_5 < 0$). O nó 4, que está na vizinha de transmissão desses nós, atualiza sua TCL, promíscuamente, a cada pacote encaminhado. Os dois fluxos são finalizados ao mesmo instante e em seguida os nós cooperativos mudam de posição, formando um quadrado com os nós 5 e 0, como ilustrado na Figura 4.3(c). Essa disposição foi definida para garantir que os fluxos desses nós sejam encaminhados apenas pelo nó 4, que é o centro do quadrado. Após essa nova disposição, são iniciados os quatro fluxos, como mostrado na Figura 4.3(d). O nó 4 utiliza o sistema de créditos e o mecanismo de encaminhamento prioritário para priorizar os fluxos dos nós com créditos positivos (nós 1 e 3) em relação ao fluxo dos nós com créditos negativos (nós 5 e 6). Os quatro fluxos são encerrados ao mesmo tempo e a simulação é finalizada em tempo suficiente para a fila de pacotes do nó 4 ser inteiramente servida.

A segunda etapa de testes utiliza o mesmo cenário, no entanto, o nó 4 envia pacotes ao nó 8 enquanto encaminha pacotes do outros quatro fluxos (ver Figura 4.3(e)). O acréscimo desse fluxo é utilizado para mostrar como o grau de comportamento cooperativo do nó 4 influencia os desempenhos dos quatro fluxos roteados.

#### 4.2.1.1 Variação do Tamanho das Filas

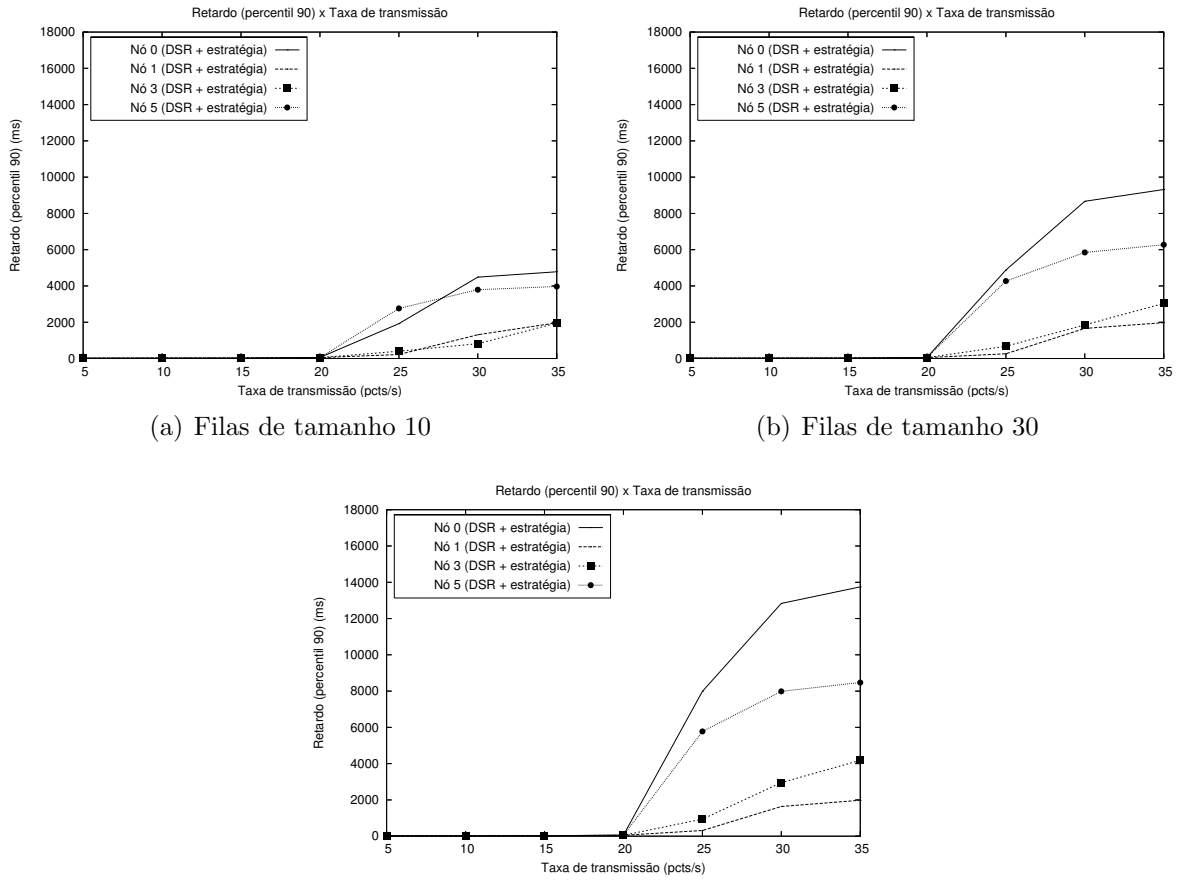
a Figura 4.4 ilustra os gráficos resultantes da variação do tamanho das filas sobre o retardo na entrega dos pacotes dos quatro nós analisados. Nos três gráficos, nota-se que o retardo obtido pelos fluxos dos nós cooperativos 1 e 3 foi menor que o dos fluxos dos nós 0 e 5, os quais não encaminharam pacotes. Neste cenário, o retardo pode ser observado apenas para as taxas superiores a 20 pacotes por segundo, visto que as taxas inferiores não foram suficientes para acumular pacotes nas filas de encaminhamento do nó 4.

A diferença do retardo obtido pelos fluxos é resultado do mecanismo de encaminhamento prioritário que prioriza o envio de pacotes dos nós com créditos positivos aos pacotes dos nós com créditos negativos.

O aumento no tamanho das filas permite um provisionamento diferenciado, inclusive, entre os fluxos de uma mesma classe (Positiva ou Negativa) devido ao fato de que a fila não é preenchida rapidamente e os pacotes de menor prioridade podem ser armazenados por mais tempo. Dessa forma, quanto maior o tamanho das filas, maior é a

vantagem obtida pelo fluxo do nó que está na subclasse *Alta* em que o peso de serviço é mais elevado. Dessa forma, o retardo sofrido pelo fluxo do nó 1 (subclasse *Alta*) é inferior ao sofrido pelo fluxo do nó 3 (subclasse *Média*), visto que o nó 1 encaminhou pacotes por mais tempo no início da simulação.

A importância de possuir créditos positivos para estabelecer conexões é reforçada com o aumento no tamanho das filas, como pode ser observado no retardo sofrido pelo fluxo do nó 0, quando utilizado filas de tamanho cinqüenta. Isso ocorre, devido ao fato de que o aumento do espaço de armazenamento melhora o desempenho da fila de maior prioridade, visto que a cada rodada do escalonador, essa fila possui pelo menos o número máximo de pacotes que podem ser servidos. Dessa forma, a vazão relativa da fila de maior prioridade é servida frequentemente em seu valor máximo, o que retarda o escalonamento das outras filas.




(a) Filas de tamanho 10

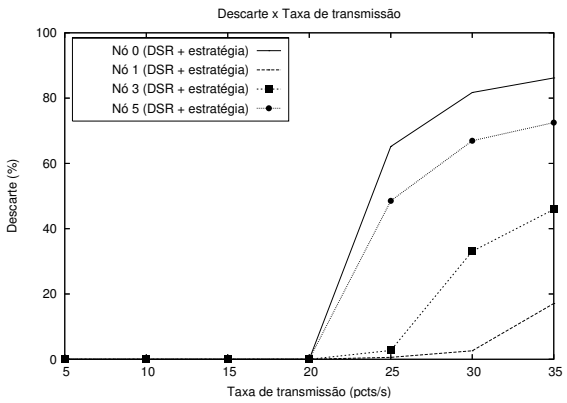
(b) Filas de tamanho 30

(c) Filas de tamanho 50

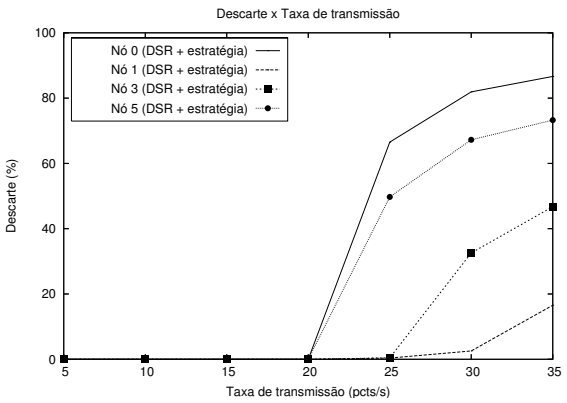
**Figura 4.4** Retardo X Taxa de Transmissão-variação do tamanho das filas de encaminhamento

A Figura 4.5 ilustra os gráficos resultantes da variação do tamanho das filas sobre

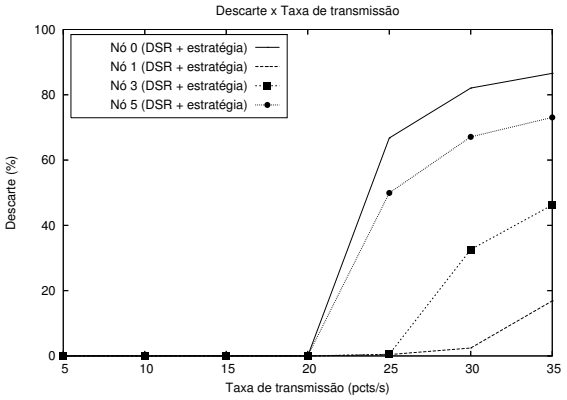
a taxa de descarte na entrega dos pacotes dos quatro nós analisados. A taxa de descarte não varia muito com o tamanho das filas, visto que a quantidade de fluxos roteados pelo nó 4 é constante e servida por uma mesma vazão máxima. Dessa forma, um tamanho maior das filas, apenas prolonga o tempo necessário para preenchê-las. Como pode ser visto nesses gráficos a estratégia foi capaz de diferenciar bem o controle da taxe de descarte de cada fluxo de acordo com a quantidade de créditos do respectivo nó fonte.




(a) Filas de tamanho 10




(b) Filas de tamanho 30

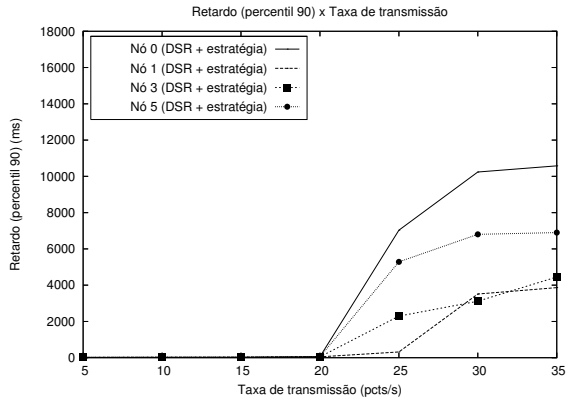



(c) Filas de tamanho 50

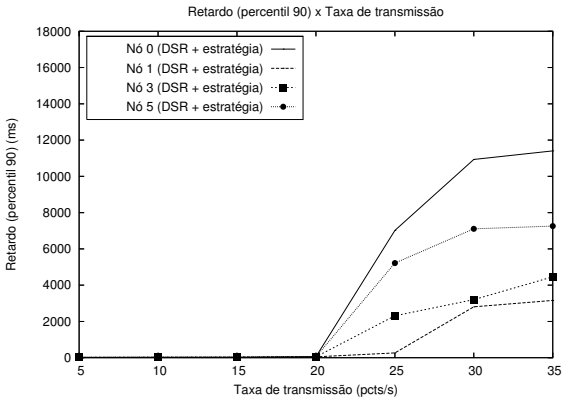
**Figura 4.5** Descarte X Taxa de Transmissão-variação do tamanho das filas de encaminhamento

**4.2.1.2 Variação dos Pesos de Encaminhamento** os gráficos da Figura 4.6 apresentam a influência dos pesos das filas de encaminhamento sobre o retardo sofrido pelos fluxos roteados. A Figura 4.6(a) ilustra o gráfico resultante da configuração $P_f1 = 4$, $P_f2 = 3$, $P_f3 = 2$ e $P_f4 = 1$. Nesse gráfico, nota-se o gerenciamento do mecanismo de encaminhamento prioritário sobre os fluxos roteados, mesmo quando utilizados pesos relativamente próximos. Semelhante a variação do tamanho das filas, os fluxos da classe

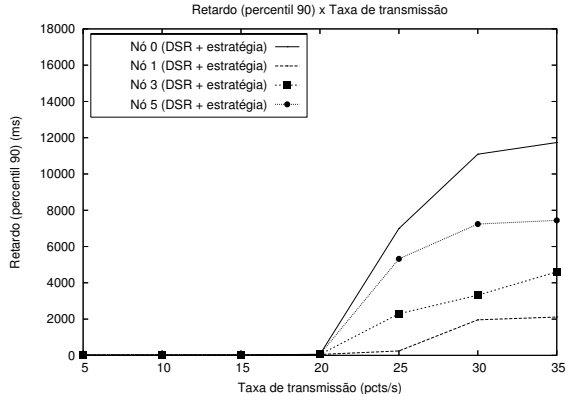
*Positiva* obtiveram vantagem sobre os fluxos da classe *Negativa*. Nessa configuração, também nota-se um retardo diferenciado entre os fluxos das subclasses de uma mesma classe, como pode ser visto no retardo, em geral, menor sofrido pelo fluxo do nó 1 se comparado ao sofrido pelo fluxo do nó 3.



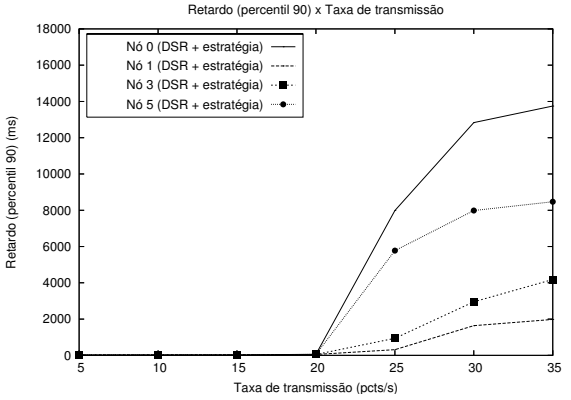

(a) $P_f1 = 4| - |P_f2 = 3| - |P_f3 = 2| - |P_f4 = 1$




(b) $P_f1 = 5| - |P_f2 = 3| - |P_f3 = 2| - |P_f4 = 1$




(c) $P_f1 = 6| - |P_f2 = 3| - |P_f3 = 2| - |P_f4 = 1$




(d) $P_f1 = 8| - |P_f2 = 3| - |P_f3 = 2| - |P_f4 = 1$

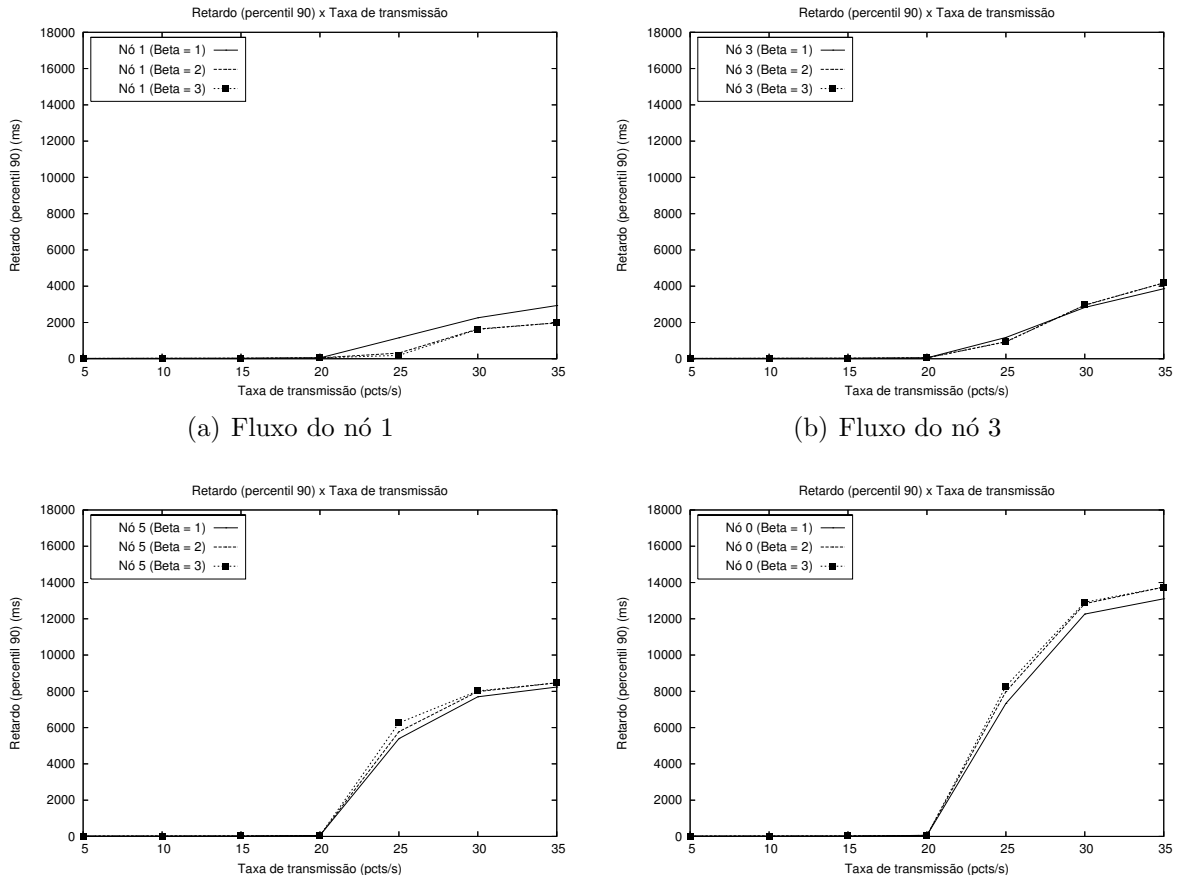
**Figura 4.6** Retardo X Taxa de Transmissão - variação dos pesos de encaminhamento

Para cada um dos gráficos, nota-se um aumento gradual da diferença entre os fluxos dos nós roteados. Isso ocorre devido ao fato que quando se aumenta a diferença entre os pesos, aumenta-se também a diferença entre a quantidade de pacotes servidos por cada fila (vazão relativa). A Figura 4.6(d) ilustra o gráfico resultante da configuração $P_f1 = 8$, $P_f2 = 4$, $P_f3 = 2$ e $P_f4 = 1$. Nesse gráfico, nota-se que a diferença entre o retardo sofrido pelos fluxos é mais espaçada. Nesse caso, como a diferença entre os pesos de encaminhamento é maior (dobrado a cada fila), é possível privilegiar ainda mais o fluxo servido pela subclasse *Alta* em relação às demais subclasses. Por outro lado, o fluxo servido pela subclasse *Baixa* sofre um retardo muito elevado que pode ser visto como

uma forma de punição à não obtenção de créditos no encaminhamento de pacotes.

#### 4.2.1.3 Variação do Valor de $\beta$

o efeito da variação de $\beta$ sobre o retardo, na entrega dos pacotes, é apresentado individualmente para cada fluxo nos gráficos da Figura 4.7.




(a) Fluxo do nó 1

(b) Fluxo do nó 3

(c) Fluxo do nó 5

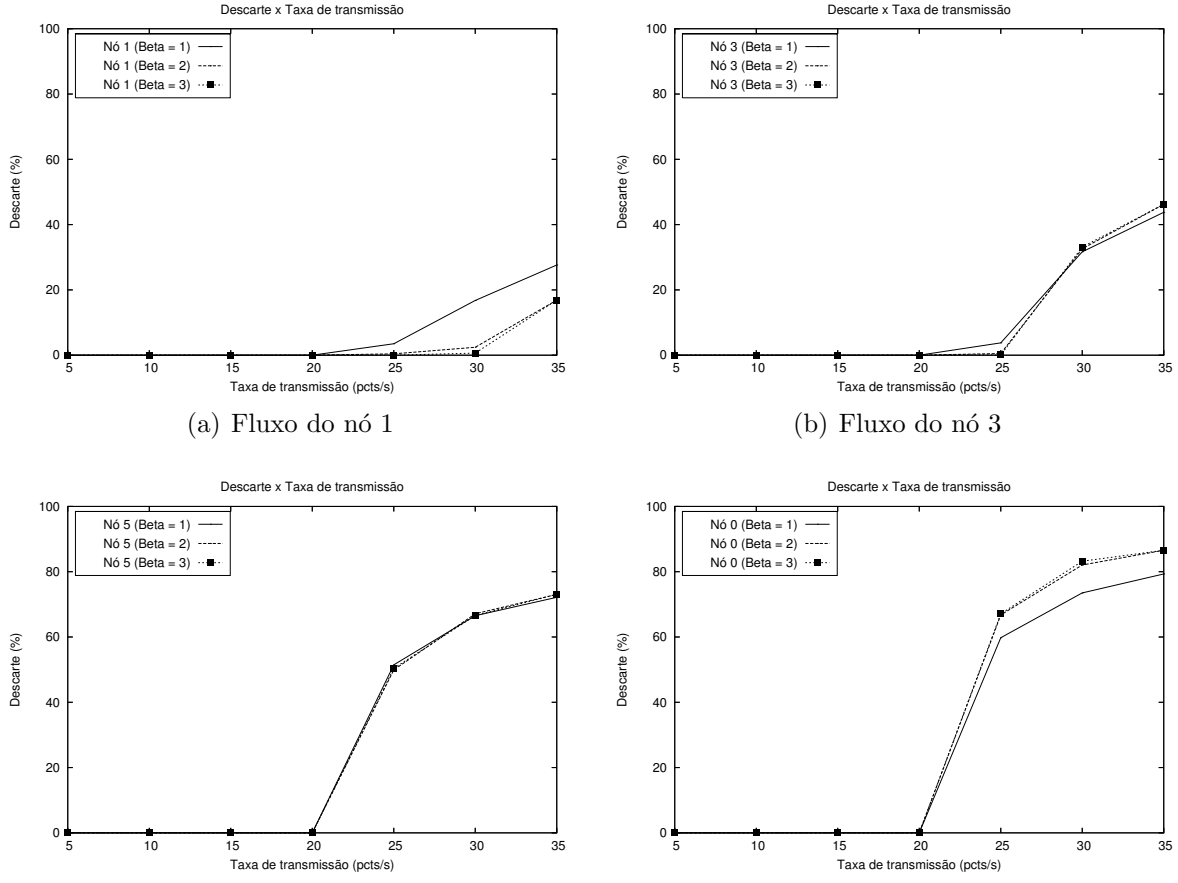
(d) Fluxo do nó 0

**Figura 4.7** Retardo X Taxa de Transmissão - variação do valor de $\beta$

No cenário utilizado, a quantidade de créditos obtidos pelos nós cooperativos na fase de encaminhamento de pacotes foi suficiente para mantê-los com créditos positivos na maior parte do tempo. Dessa forma, nota-se que o retardo do fluxo de um nó cooperativo variou pouco com relação aos valores de $\beta$. Quando $\beta$ assumiu o valor 1, o retardo sobre o fluxo do nó 1 foi um pouco pior, visto que nesse caso, esse nó chegou a ficar com créditos negativos no final da simulação. O mesmo não ocorreu com o nó 3 pelo fato da sua subclasse de serviço (*Média*) ser servida em média com a metade da vazão relativa da subclasse do nó 1. No caso do nó 0, ocorreu o inverso em relação ao valor de $\beta$ ser 1,

visto que no período final da simulação o fluxo do nó 1 foi classificado na classe *Negativa* de serviço, o que diminuiu a disparidade da vazão relativa. Caso o tempo de simulação fosse estendido, a provável permanência dos quatro fluxos na classe *Negativa* ocasionaria um aumento no retardo médio sobre os fluxos dos nós cooperativos.

Os gráficos da Figura 4.8 apresentam o efeito da variação de $\beta$ sobre a taxa de descarte de cada um dos quatro fluxos. Semelhante ao ocorrido com o retardo, nota-se uma variação apenas sobre os fluxos dos nós 1 e 0. Quando $\beta$ é configurado com o valor 1 a taxa de descarte do nó 1 sofre um aumento, visto que esse fluxo para a ser servido por uma subclasse *Negativa* no final da simulação. Por outro lado, a taxa de descarte do fluxo do nó 0 diminui, visto que a disparidade na classificação dos fluxos foi menor no final da simulação.




(a) Fluxo do nó 1

(b) Fluxo do nó 3

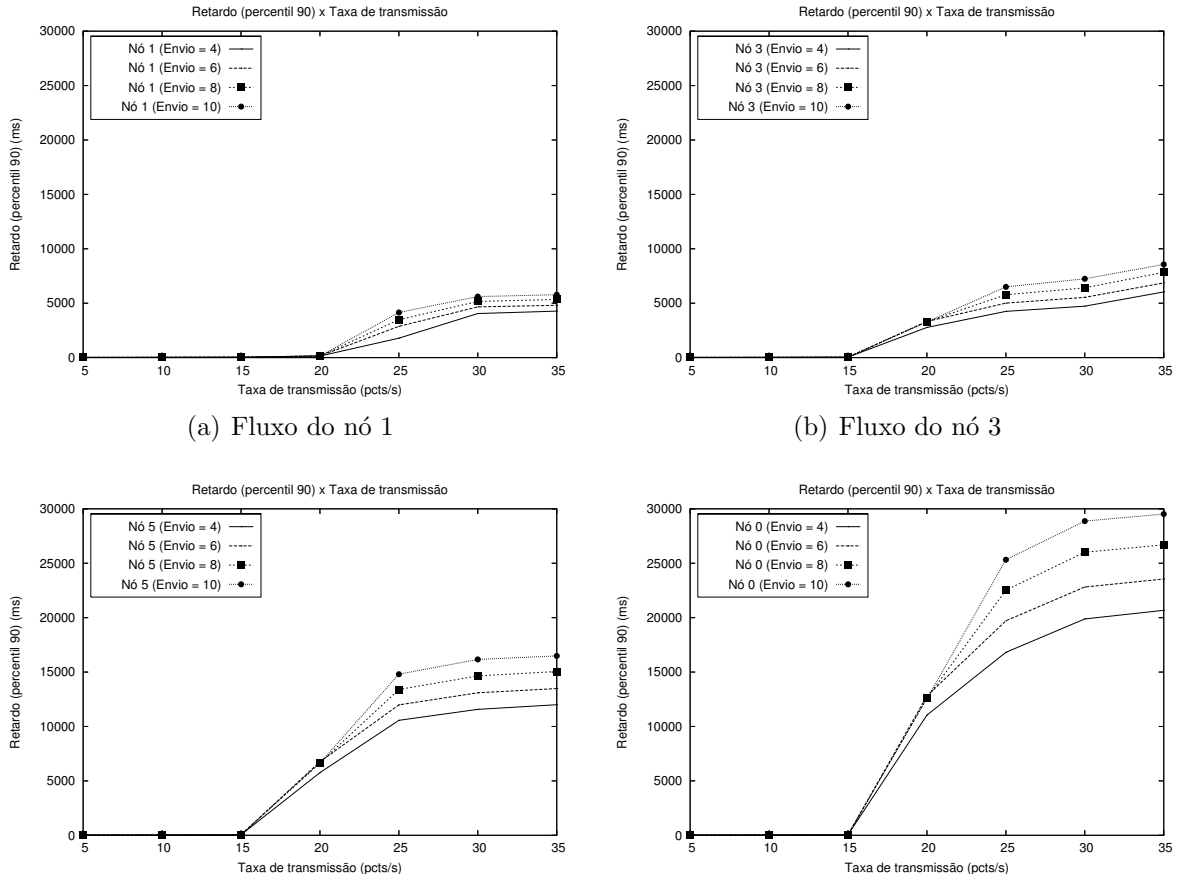
(c) Fluxo do nó 5

(d) Fluxo do nó 0

**Figura 4.8** Descarte X Taxa de Transmissão - variação do valor de $\beta$

**4.2.1.4 Variação do Peso de Envio** na segunda etapa dos testes de calibragem, foi analisada a influência da variação do peso de envio sobre o retardo e a taxa de descarte dos fluxos roteados pelo nó 4. Essa variação define o grau de comportamento cooperativo do nó 4. Nesse caso, a fila do nó 4 escalona 5 fluxos sendo quatro de encaminhamento e uma de envio.

A Figura 4.9 ilustra, para cada fluxo, como a variação do peso de envio influencia o retardo na entrega dos pacotes. Em cada gráfico, observa-se que quando o nó 4 aumenta a prioridade de envio do seu fluxo, o retardo dos fluxos encaminhados também aumenta, como era esperado visto que a vazão relativa das filas de encaminhamento diminuiu. Nota-se também que o mecanismo de encaminhamento prioritário continua funcionando de forma eficaz mesmo quando o nó 4 aumenta a prioridade de serviço do seu fluxo. É evidente que no caso em que o nó 4 prioriza apenas o envio de seu fluxo, o comportamento desse nó é semelhante ao de um nó egoísta.




(a) Fluxo do nó 1

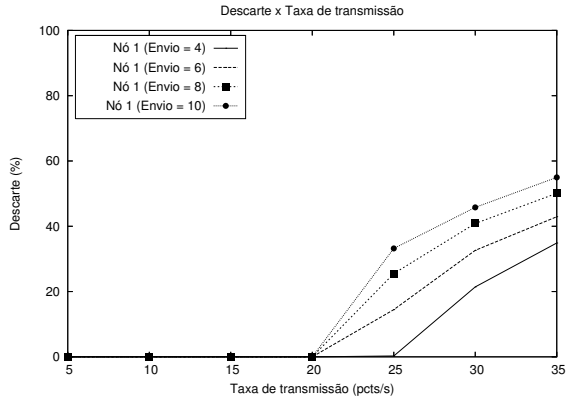
(b) Fluxo do nó 3

(c) Fluxo do nó 5

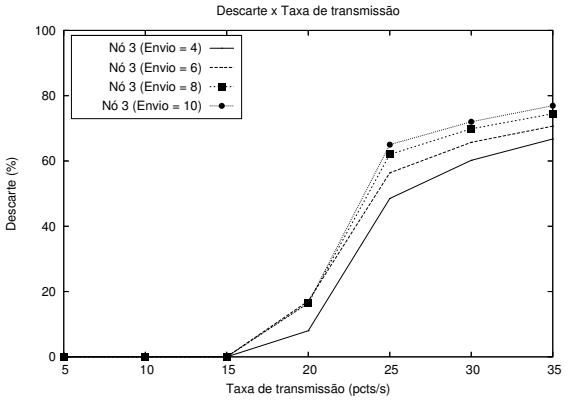
(d) Fluxo do nó 0

**Figura 4.9** Retardo X Taxa de Transmissão - variação do peso de envio

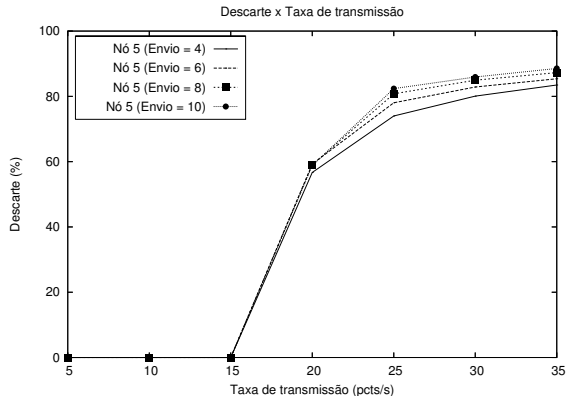
A Figura 4.10 ilustra, para cada fluxo, como a variação do peso de envio influencia na taxa de descarte. A variação é mais intensa sobre os fluxos dos nós cooperativos pelo fato do aumento na prioridade do fluxo de envio representar uma maior concorrência sobre os fluxos servidos pela classe *Positiva*. Dessa forma, o serviço dessas classes permaneceu constante, enquanto a vazão relativa das filas de encaminhamento diminuiu. A taxa de descarte dos fluxos servidos pela classe *Negativa* não foi muito atingida, pelo fato de que a variação da vazão relativa não foi suficiente para provocar danos ao já comprometido peso de serviço. Dessa forma, as filas da classe *Negativa* apenas são preenchidas um pouco mais rápido e após esse momento a taxa de descarte mantém-se constante.



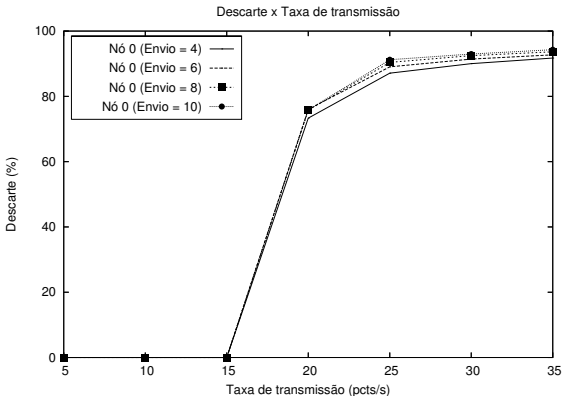

(a) Fluxo do nó 1




(b) Fluxo do nó 3




(c) Fluxo do nó 5




(d) Fluxo do nó 0

**Figura 4.10** Descarte X Taxa de Transmissão - variação do peso de envio

### 4.2.2 Testes de Avaliação

Os testes de avaliação utilizam a configuração base dos testes de calibragem para analisar a estratégia em um cenário mais complexo. Nesses testes, foram simulados e analisados os comportamentos: cooperativo, egoísta intencional e egoísta não intencional. Vale ressaltar que a calibragem da estratégia foi realizada através de uma análise empírica dos resultados, portanto, não é possível garantir que essa configuração seja ótima.

O cenário de simulação é constituído de 36 nós, em que parte move-se de forma aleatória (modelo *random way-point*) em um plano de 900x600m, como ilustrado na Figura 4.11. Foram definidos três nós fonte (e.g., $F_1$, $F_2$ e $F_3$) e três nós destino (e.g., $D_1$, $D_2$ e $D_3$) para geração e consumo de tráfego CBR. Esses nós são estáticos e posicionados nas extremidades da rede.

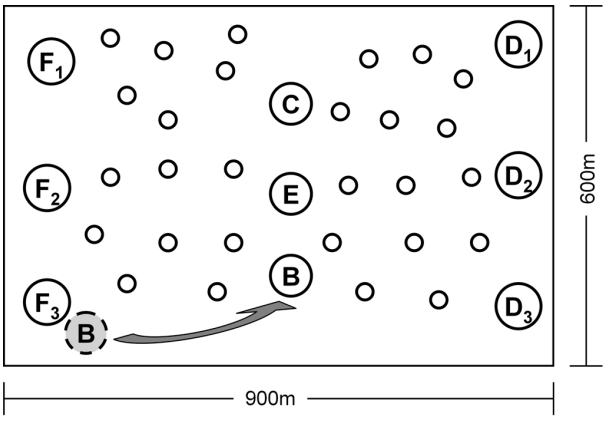



**Figura 4.11** Cenário dos testes de avaliação

O posicionamento fixo e estático dos nós fonte e destino é motivado para possibilitar uma avaliação mais coerente dos resultados, visto que quando se utiliza origens e destinos móveis, corre-se o risco da comunicação direta entre eles ser possível, o que dificulta a análise do comportamento dos nós intermediários. Os nós $C$, $E$ e $B$ (alvos da análise) representam, respectivamente, um nó cooperativo, um nó egoísta intencional e um nó egoísta não intencional (inicialmente o nó $B$ encontra-se na borda rede). Esses nós são móveis, no entanto, sua locomoção está limitada ao centro da rede, região em que é verificada uma maior freqüência de pacotes encaminhados [51]. Esses nós também geram tráfego CBR que é destinado a um mesmo nó destino (nó $D_2$). Dessa forma, a distância média, em saltos, ao nó destino $D_2$ é equivalente para os três fluxos analisados e aumenta-se a possibilidade desses fluxos serem roteados por um mesmo nó intermediário.

A configuração dos testes de avaliação é apresentada na Tabela 4.2. Para cada ponto de apresentação (taxa de transmissão) são executadas vinte rodadas da simulação. O tempo de cada simulação foi de 1800 segundos em que foi possível analisar o estado consistente com um bom nível de confiança. Em cada rodada, são utilizadas sementes de geração de números aleatórios diferentes para o ambiente de simulação e para a geração da movimentação dos nós intermediários. A movimentação desses nós foi gerada através da ferramenta *BonnMotion* [57] que é capaz de exportar os cenários para o NS-2. A movimentação dos nós ocorre sempre em linha reta (dois pontos do plano) e muda de sentido e direção após um tempo aleatório de pausa em que os nós permanecem parados. Para cada cenário também foram geradas movimentações independentes para os nós analisados (e.g., $C$ $E$ e $B$), em que a ferramenta *BonnMotion* foi configurada com parâmetros diferentes para limitar a movimentação dos nós nas imediações do centro da rede. Os nós se movimentam a uma velocidade constante de $1m/s$, que é equivalente à velocidade de uma pessoa andando. Os gráficos gerados possuem barras de erro, em torno das médias das vinte replicações, utilizando um intervalo de confiança de 95%.

**Tabela 4.2** Configuração da simulação - testes de avaliação

(a) Parâmetros da simulação

| **Parâmetro** | **Valor** |
|---|---|
| Cenário (área) | 900m x 600m |
| Alcance do Sinal | 250m |
| Movimento | *Random Way-Point* |
| Velocidade | 1m/s |
| Tempo de Pausa | Aleatório |
| MAC | IEEE 802.11b (DCF, 2Mbps) |
| Antena | Omnidirecional e Simétrica |
| Aplicação | CBR |
| Transporte | UDP |
| Tamanho do Pacote | 512 B |
| Taxa (pcts/s) | 5, 7.5, 10, 12.5, 15, 17.5, 20 |
| Tempo de Simulação | 1800 s |
| Número de rodadas | 20 |
| Intervalo de Confiança | 95% |

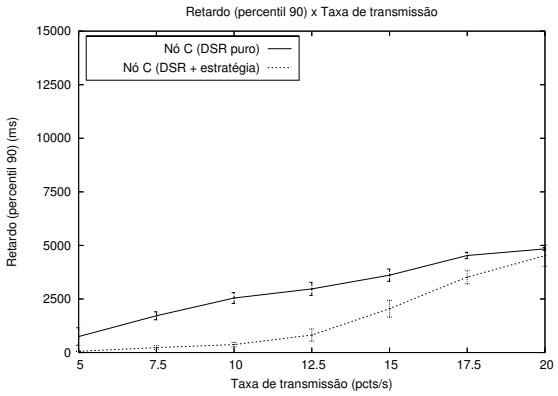
(b) Parâmetros da estratégia

| **Parâmetro** | **Valor** |
|---|---|
| $\beta$ | 2 |
| $\alpha$ | 1 |
| $P_s$ | 8 |
| $P_f1$ | 8 |
| $P_f2$ | 4 |
| $P_f3$ | 2 |
| $P_f4$ | 1 |
| $T_f$ | 50 (Pacotes) |

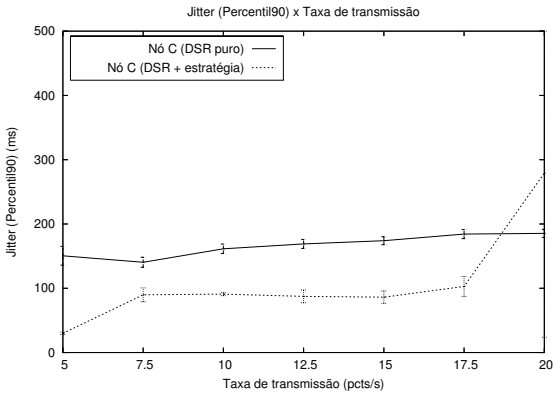
A simulação é estruturada segundo uma seqüência de passos. Inicialmente, tem-

se apenas os fluxos dos nós origem $F_1$, $F_2$ e $F_3$ aos respectivos nós destinos $D_1$, $D_2$ e $D_3$. Nessa fase, os nós $C$ e $E$ são requisitados ao encaminhamento de pacotes, enquanto o nó $B$ está isolado dessa função na borda da rede. Dessa forma, apenas o nó $C$ está adquirindo créditos, visto que o nó $E$ é egoísta na função de encaminhar pacotes. Em um segundo momento, o nó $B$ se dirige ao centro da rede e movimenta-se na mesma região dos nós $C$ e $E$. Em seguida, são iniciados os tráfegos para cada um desses nós em direção ao nó destino $D_2$. A análise sobre os fluxos dos nós $C$, $E$ e $B$ é iniciada após o estabelecimento de um estado consistente sobre essas conexões (intervalo de 5% do tempo total de simulação).

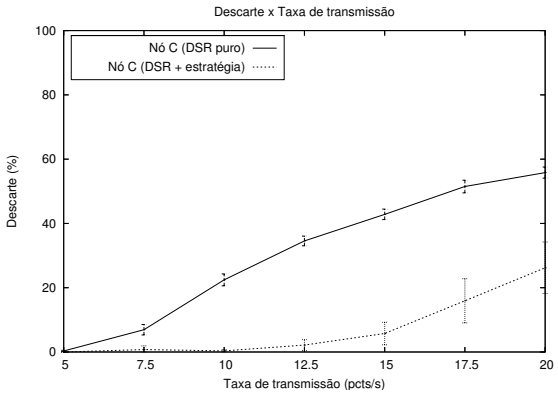
**4.2.2.1 Fluxo do Nó Cooperativo** a Figura 4.12 apresenta os gráficos resultantes da análise do fluxo do nó cooperativo (nó C). Para cada métrica analisada o fluxo do nó cooperativo foi comparado entre as implementações *pura* e *modificada* do DSR.




(a) Retardo x Taxa de Transmissão




(b) *Jitter* x Taxa de Transmissão




(c) Descarte x Taxa de Transmissão

**Figura 4.12** Análise do fluxo do nó cooperativo

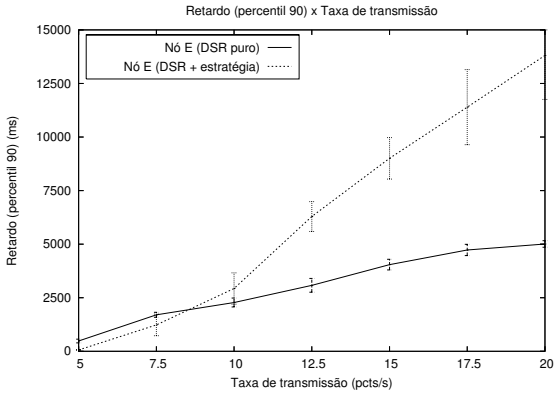
O gráfico do retardo, ilustrado pela Figura 4.12(a), mostra que quando utilizada a estratégia proposta, o fluxo do nó cooperativo obteve um retardo inferior para as taxas apresentadas. Esse resultado era esperado, visto que a fila de pacotes do *DSR puro* é do tipo FIFO sem qualquer política de escalonamento. Dessa forma, os pacotes são servidos por ordem de chegada, independente do nó origem ser egoísta ou cooperativo. Por outro lado, o mecanismo de escalonamento proposto neste trabalho serve os fluxos de acordo com a quantidade de créditos dos respectivos nós origem. Dessa forma, o nó cooperativo obteve um serviço mais privilegiado pelo fato de ter adquirido créditos antes e durante o envio de seus pacotes. Pode-se verificar no gráfico que à medida que a taxa de transmissão aumenta, o desempenho da estratégia proposta torna-se semelhante ao *DSR puro*. Isso é devido a fato do aumento das taxas de transmissão congestionar as filas de encaminhamento.

O gráfico da Figura 4.12(b) apresenta o resultado do *jitter* na entrega de pacotes do nó cooperativo. Nota-se um desempenho melhor da estratégia até a taxa 17.5 pacotes/s, em que houve um aumento considerável no retardo devido ao congestionamento da rede.

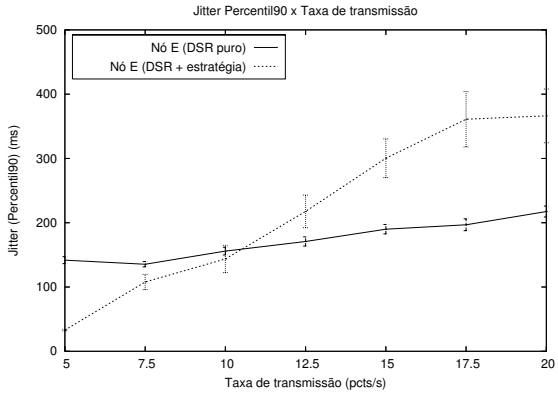
O gráfico da Figura 4.12(c) apresenta a taxa de descarte resultante na entrega de pacotes do nó cooperativo. Para as duas implementações do DSR, nota-se o aumento da taxa de descarte com o aumento da taxa de transmissão, visto que os nós intermediários têm suas filas congestionadas com o aumento na requisição por encaminhamento de pacotes. O bom desempenho da estratégia, no privilegio ao fluxo do nó cooperativo, era esperado visto que a vazão relativa de pacotes desse fluxo é superior ao oferecido pelo *DSR puro*, que divide a vazão igualmente entre os fluxos roteados.

**4.2.2.2 Fluxo do Nó Egoísta** os gráficos do retardo fim-a-fim, do *jitter* e da taxa de descarte obtidos pelo fluxo do nó egoísta (nó E), são ilustrados na Figura 4.13. Nota-se que o fluxo do nó egoísta, quando utilizado o *DSR + estratégia*, obteve um serviço, em geral, pior do que o obtido quando utilizado o *DSR puro*. Isso se deve ao fato do *DSR puro* não fazer distinção entre os nós no encaminhamento de seus pacotes.

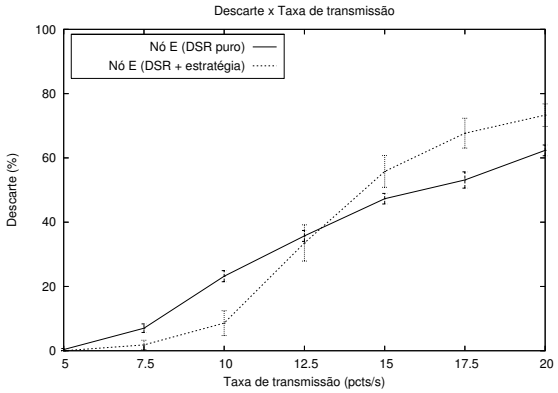
O gráfico do retardo, ilustrado pela Figura 4.13(a), mostra que quando utilizada a estratégia proposta, o fluxo do nó egoísta obteve um retardo superior para a maioria das taxas apresentadas. Inicialmente, o retardo do nó egoísta é um pouco melhor devido à possibilidade dos fluxos dos três nós fonte $F_1$, $F_2$ e $F_3$ estarem disputando as mesmas filas




(a) Retardo x Taxa de Transmissão




(b) *Jitter* x Taxa de Transmissão




(c) Descarte x Taxa de Transmissão

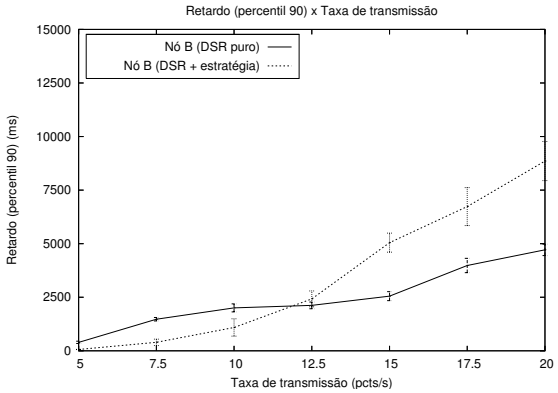
**Figura 4.13** Análise do fluxo do nó egoísta

dos nós intermediários. Enquanto o *DSR puro* divide a vazão igualmente entre os fluxos, o *DSR + estratégia* serve melhor o fluxo do nó egoísta em relação aos fluxos dos nós fonte que estão com a quantidade de créditos mais negativa. No entanto, com o aumento das taxas de transmissão, a punição imposta pela estratégia (baixa vazão relativa) aos fluxos dos nós com quantidade de créditos negativa é mais intensa. Dessa forma, o retardo na entrega de pacotes do fluxo do nó egoísta aumenta rapidamente.

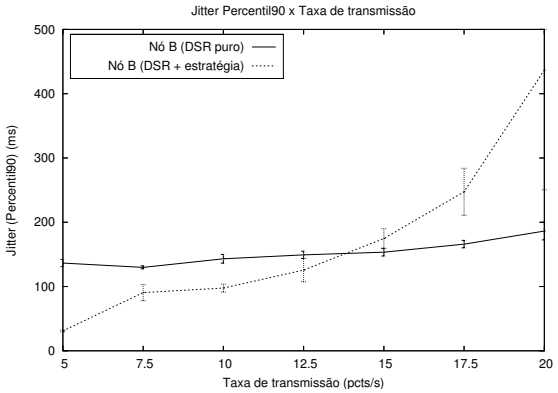
A análise da variação da taxa de descarte, Figura 4.13(c), e do *jitter*, Figura 4.13(b), segue a mesma lógica da análise da variação do retardo, em que o aumento da taxa de transmissão eleva o nível de punição aos fluxos dos nós que estão com créditos negativos.

**4.2.2.3 Fluxo do Nó de Borda** a Figura 4.14 ilustra os gráficos do retardo fim-a-fim, *jitter* e taxa de descarte resultantes da análise do fluxo do nó de borda (nó B). Em

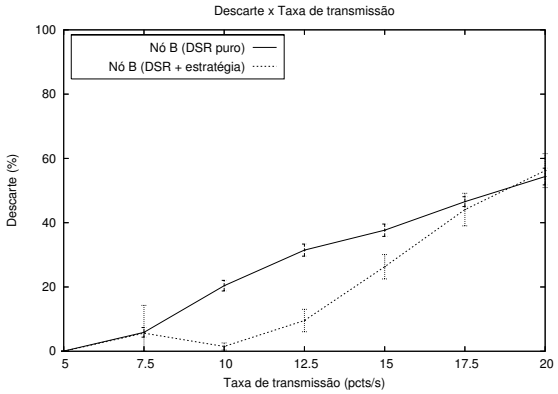
todos os gráficos, nota-se que o nó B foi punido pela falta de créditos quando utilizada a estratégia, no entanto, essa punição foi mais branda do que a imposta ao nó egoísta. Isso se deve ao fato de que nó B passa a adquirir créditos quando se dirige ao centro da rede. Dessa forma, o seu fluxo é servido por uma classe mais prioritária que o fluxo do nó egoísta. No caso desse cenário e da configuração da estratégia utilizada, o comportamento cooperativo do nó B não foi suficiente para tornar sua quantidade de créditos positiva, na TCL dos nós intermediários. No entanto, caso a simulação fosse mais longa, o nó B ficaria com a quantidade de créditos positiva, visto que encaminha mais pacotes do que envia (a soma dos pesos das filas de encaminhamento - 15, é maior do que o peso da fila de envio - 8) e o valor de $\beta$ é o dobro do valor de $\alpha$.




(a) Retardo x Taxa de Transmissão




(b) *Jitter* x Taxa de Transmissão

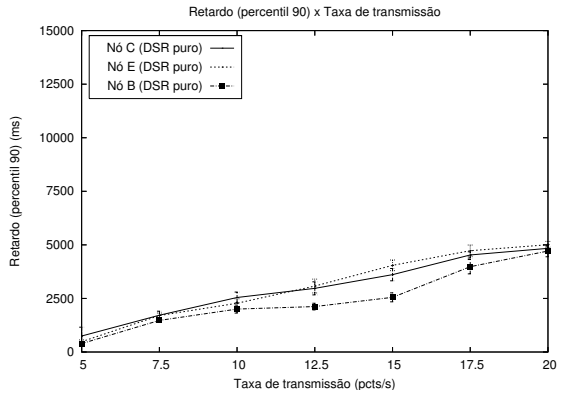



(c) Descarte x Taxa de Transmissão

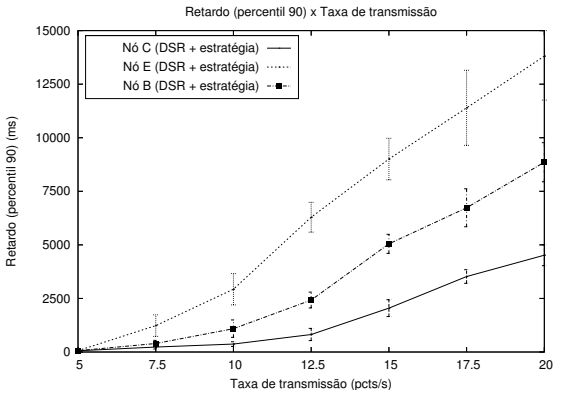
**Figura 4.14** Análise do fluxo do nó de borda

**4.2.2.4 DSR puro *Versus* DSR + estratégia** a Figura 4.15 ilustra os gráficos do retardo fim-a-fim, do *jitter* e da taxa de descarte dos fluxos analisados, agrupados pelo

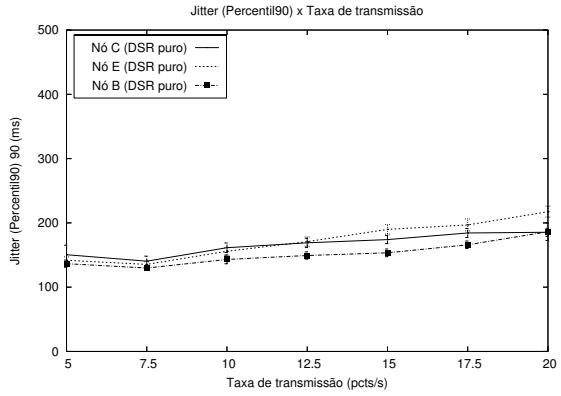
tipo de implementação.



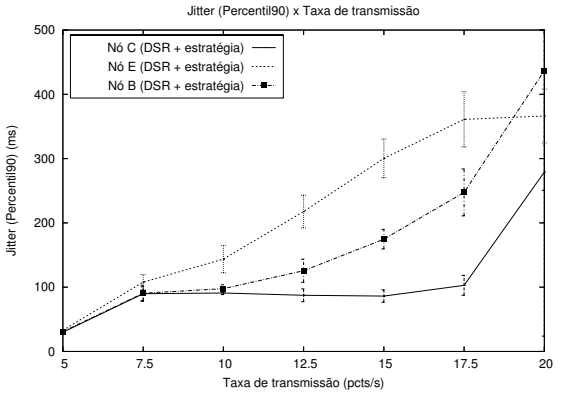

(a) Retardo - DSR puro



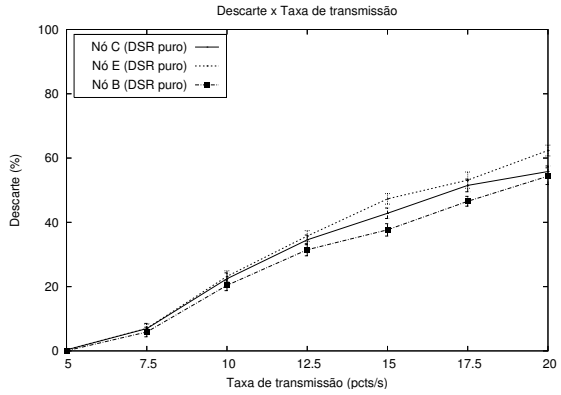

(b) Retardo - DSR + estratégia



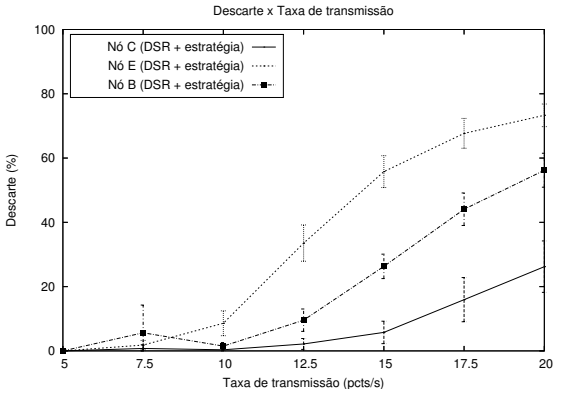

(c) *Jitter* - DSR puro




(d) *Jitter* - DSR + estratégia




(e) Descarte - DSR puro




(f) Descarte - DSR + estratégia

**Figura 4.15** DSR puro x DSR + estratégia

Através desses gráficos é possível constatar que o *DSR puro* trata os fluxos roteados de forma semelhante, visto que os resultados estão muito próximos. Por outro lado, é possível verificar que a estratégia proposta diferencia bem os três fluxos. O fluxo do nó

C obteve melhores resultados em relação aos fluxos dos outros dois nós e em relação ao seu desempenho quando utilizado o *DSR puro*. O fluxo do nó B foi punido, no entanto, essa punição é mais branda do que a imposta ao nó egoísta, que em nenhum momento adquire créditos através do encaminhamento de pacotes. Dessa forma, pode-se verificar que a estratégia privilegia o fluxo do nó cooperativo, enquanto pune os fluxos dos nós que possuem créditos negativos, visto que a QoS desses foi inferior à provida pelo *DSR puro*.

# CAPÍTULO 5

# CONCLUSÃO

Este trabalho apresentou uma estratégia autônoma de incentivo à cooperação baseada em créditos. Seu objetivo é o combate ao comportamento egoísta através de incentivo por: privilégios, encaminhamento prioritário dos pacotes dos nós cooperativos, e punições aos fluxos dos nós com créditos negativos. Para isso, foi desenvolvido um sistema de créditos e um mecanismo de encaminhamento prioritário de pacotes que utilizam apenas informações locais e não são subordinados a uma entidade central de gerenciamento.

Os experimentos mostraram que a estratégia é eficaz, visto que controla de forma autônoma os incentivos, privilegia os fluxos dos nós cooperativos e pune os fluxos dos nós com quantidade de créditos negativa. A eficiência foi avaliada através de comparações entre o desempenho do protocolo de roteamento DSR adaptado pela estratégia e o sem modificações. Os testes mostraram que o protocolo DSR, quando munido da estratégia, é capaz de diferenciar os serviços dos fluxos de acordo com o comportamento dos respectivos nós fonte, enquanto o *DSR puro* trata os fluxos igualmente.

O restante deste capítulo apresenta as possíveis investigações futuras deste trabalho na Seção 5.1.

## 5.1 SUGESTÕES DE INVESTIGAÇÕES FUTURAS

Como sugestões para trabalhos futuros podem-se tratar os seguintes casos identificados durante a produção deste trabalho:

- **Alcance do Gerenciamento:** a quantidade de créditos de um determinado nó é gerenciada apenas por uma porção da rede. Isso ocorre quando há pouca mobilidade na rede em que a vizinhança de um nó permanece constante por um grande intervalo de tempo. Para tornar a estratégia mais justa, faz-se necessário o compartilhamento das informações de créditos. No entanto, esse compartilhamento de

informações exige o uso de medidas de segurança na autenticação e integridade das comunicações.

- **Custo-benefício do comportamento cooperativo:** o escalonador cíclico desenvolvido define o grau do comportamento cooperativo dos nós, no entanto, essa definição mantém-se constante e não é resultante das métricas de desempenho dos nós. Dessa forma, faz-se necessário o uso de um mecanismo dinâmico em que os interesses dos nós sejam avaliados. Através do uso de funções de utilidade, como nas estratégias baseadas em teoria dos jogos, um nó pode controlar seu comportamento cooperativo baseado nas suas métricas de desempenho e no nível corrente de seus recursos.

- **Heterogeneidade dos nós:** neste trabalho, a quantidade de recursos de um nó (capacidade cooperativa) não é analisado pelo sistema de contabilização de créditos, portanto, o trabalho de um PDA 100% cooperativo é contabilizado da mesma forma que o trabalho de um *notebook* 100% cooperativo. Para aumentar a justiça na rede, os trabalhos de Urpi et al. [31] e Srinivasan et al. [33] apresentam um modelo matemático em que a capacidade cooperativa dos nós é utilizada na contabilização dos *payoffs*. Como esse modelo é capaz de representar as características de uma MANET, ele pode ser adaptado à estratégia proposta.

- **Estabelecimento de Rotas:** a contabilização dos créditos durante o processo de estabelecimento de rotas também é executada neste trabalho e os valores de $\alpha$ e $\beta$ da equação 3.1 são mantidos os mesmos do processo de comunicação. No entanto, como a comunicação entre os nós é dependente do processo de estabelecimento de rotas, a oferta de incentivos ainda maiores para os nós cooperativos é apropriada, exigindo uma atualização dos valores de $\alpha$ e $\beta$. Ainda sobre o processo de descoberta de rotas, os algoritmos de roteamento podem utilizar as informações da TCL como parâmetro de entrada para escolha das melhores rotas (e.g., rotas em que o nó origem possui quantidade positiva de créditos, rotas em que os nós intermediários foram servidos pelo nó origem, dentre outras). Para isso, a coluna de créditos da TCL seria dividia em duas: uma para a contabilização de créditos no caso em que o nó faz parte da rota e outra para contabilização no caso em que o nó não faz parte da rota. Dessa forma, é possível identificar exatamente para quais nós um determinado nó já utilizou ou prestou serviços.

- **Comportamento Malicioso:** o simples fato de um nó malicioso não creditar os

pacotes encaminhados pelos seus vizinhos é insuficiente para influenciar o comportamento dos outros nós da rede, visto que as informações da TCL não são compartilhadas. No entanto, como essa estratégia é baseada no uso de identificadores, o mecanismo de encaminhamento prioritário de pacotes pode ser burlado pelo ataque *Sybil* [36], em que, no contexto deste trabalho, um nó malicioso descobre e utiliza os identificadores dos nós cooperativos. Neste trabalho, o comportamento malicioso não foi tratado, no entanto, acreditamos que a proposta pode trabalhar em conjunto com um Sistema de Detecção de Intrusão (SDI) que combate este tipo de problema.

# REFERÊNCIAS BIBLIOGRÁFICAS


[1] C.E. Perkins et al. *Ad hoc networking*. Addison-Wesley Boston, 2001.

[2] S. Basagni. *Mobile Ad Hoc Networking*. Wiley-IEEE, 2004.

[3] L. Buttyan and J.P. Hubaux. *Security and Cooperation in Wireless Networks: Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing*. Cambridge Press (to appear), On-line: http://secowinet.epfl.ch.

[4] A. Boukerche. *Handbook of Algorithms for Wireless Networking and Mobile Computing*. Chapman & Hall/CRC, 2005.

[5] M. Conti, E. Gregori, and G. Maselli. Cooperation issues in mobile ad hoc networks. *Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on*, pages 803–808, 2004.

[6] C. E. Perkins and E. M. Royer. Ad-Hoc On Demand Distance Vector Routing. In *IEEE WMCSA'99*, pages 90–100. 1999.

[7] C. E. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *ACM Conf. on Communications Architecture, Protocols and Applications*, pages 234–244. 1994.

[8] V. Park and M. Corson. Temporally-Ordered Routing Algorithm (TORA) Version 1 -Functional Specification. Technical report, MANET Working Group, 1999.

[9] D.B. Johnson and D.A. Maltz. Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 353:153–181, 1996.

[10] N. Jindal, U. Mitra, and A. Goldsmith. Capacity of ad-hoc networks with node cooperation. *Proc. IEEE Int. Symp. Inform. Theory*, page 271, 2004.

[11] J. Li, C. Blake, D.S.J. De Couto, H.I. Lee, and R. Morris. Capacity of Ad Hoc Wireless Networks. *Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 61–69, 2001.

[12] M. D. Ortiz, A. S. C. Aguiar, D. A. Lima, M. P. Fernandez, and J. N. Souza. Incentivando a Cooperação em Redes Ad Hoc . In *Euro American Association on Telematics and Information Systems (EATIS'2007)*, maio 2007.

[13] M. D. Ortiz, A. S. C. Aguiar, D. A. Lima, M. P. Fernandez, and J. N. Souza. Análise, Implementação e Teste de uma Estratégia Autônoma de Incentivo à Cooperação em Redes Ad-Hoc. In *XXV Simpósio Brasileiro de Redes de Computores (SBRC'2007)*, pages 235–248. Belém, Brasil, maio 2007.

[14] B. O'Hara and G. Ennis. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *ANSI/IEEE Std*, 802, 1999.

[15] SIG Bluetooth. Specification of the Bluetooth system, Profiles. *V1*, 1:22, 2001.

[16] R. Hekmat. *Ad-hoc Networks:: Fundamental Properties and Network Topologies*. Springer, 2006.

[17] I. Chlamtac, M. Conti, and J.J.N. Liu. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks*, 1(1):13–64, 2003.

[18] K. Lorincz et al. Sensor networks for emergency response: challenges and opportunities. *Pervasive Computing, IEEE*, 3(4):16–23, 2004.

[19] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh. Monitoring volcanic eruptions with a wireless sensor network. *Wireless Sensor Networks, 2005. Proceeedings of the Second European Workshop on*, pages 108–120.

[20] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *Communications Magazine, IEEE*, 40(8):102–114, 2002.

[21] S.L. Wu and Y.C Tseng. *Wireless Ad Hoc Networking: Personal-area, Local-area, and the Sensory-area Networks*. Auerbach; Taylor & Francis distributor, 2007.

[22] M. Felegyhazi, J.P. Hubaux, and L. Buttyan. Cooperative Packet Forwarding in Multi-Domain Sensor Networks. *Proceedings of the First International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSeNS)*, 2005.

[23] L. Buttyan, T. Holczer, and P. Schaffer. Spontaneous Cooperation in Multi-Domain Sensor Networks. *LECTURE NOTES IN COMPUTER SCIENCE*, 3813:42, 2005.

[24] K. Mase, Y. Wada, N. Mori, K. Nakano, M. Sengoku, and S. Shinoda. Flooding schemes for a universal ad hoc network. *Industrial Electronics Society (IECON'2000)-26th Annual Confjerence of the IEEE*, 2, 2000.

[25] P. Mohapatra, J. Li, and C. GuiS. QoS in mobile ad hoc networks. *IEEE Wireless Communications*, 10(3):44–52, 2003.

[26] T.B. Reddy, I. Karthigeyan, B.S. Manoj, and C.S.R. Murthy. Quality of service provisioning in ad hoc wireless networks: a survey of issues and solutions. *Ad Hoc Networks*, 4:83–124, 2006.

[27] S.B. Lee, G.S Ahn, X. Zhang, and A.T. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60(4):374–406, 2000.

[28] B.G. Chun and M. Baker. Evaluation of packet scheduling algorithms in mobile ad hoc networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 6(3):36–49, 2002.

[29] R. Axelrod. *Evolution of Cooperation*. HarperCollins Canada, Limited, 1985.

[30] I.D. Chase. Cooperative and Noncooperative Behavior in Animals. *The American Naturalist*, 115(6):827–857, 1980.

[31] A. Urpi, M. Bonuccelli, and S. Giordanos. Modeling Cooperation in Mobile Ad Hoc Networks: a Formal Description of Selfishness. *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, 2003.

[32] F.H.P. Fitzek and M. Katz, editors. *Cooperation in Wireless Networks: Principles and Applications – Real Egoistic Behavior is to Cooperate!* ISBN 1-4020-4710-X. Springer, Abril 2006.

[33] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, and R.R. Rao. Cooperation in wireless ad hoc networks. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, 2.

[34] P. Michiardi and R. Molva. Core: A COllaborative REputation Mechanism to Enforce Node Cooperation in Mobile Ad Hoc Networks. *Sixth IFIP conference on security communications, and multimedia (CMS 2002), Portoroz, Slovenia*, 2002.

[35] M. Conti, E. Gregori, and G. Maselli. Reliable and Efficient Forwarding in Ad Hoc Networks. *Elsevier Journal of Ad Hoc Networks*, 4(3):398–415, 2006.

[36] J.R. Douceur. The Sybil Attack. *Proceedings for the 1st International Workshop on Peer-to-Peer Systems (IPTPS'02)*, 2002.

[37] S. Buchegger and J.Y Le Boudee. Self-policing mobile ad hoc networks by reputation systems. *Communications Magazine, IEEE*, 43(7):101–107, 2005.

[38] P. Obreiter and M. Klein. Vertical integration of incentives for cooperation-interlayer collaboration as a prerequisite for effectively stimulating cooperation in ad hoc networks. *Second Mediterranean Workshop on Ad-Hoc Networks (MEDHOC NET 2003), Mahdia, Tunisia*, 2003.

[39] B. Yang, T. Condie, S. Kamvar, and H. Garcia-Molina. Non-Cooperation in Competitive P2P Networks. *Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on*, pages 91–100, 2005.

[40] M. Felegyhazi, J. Hubaux, and L. Buttyan. Nash Equilibria of Packet Forwarding Strategies in Wireless Ad Hoc Networks. *Mobile Computing, IEEE Transactions on*, 5(5):463–476, 2006.

[41] S. Marti, Tj. Giuli, K. Lai, and M. Baker. Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 255–265, 2000.

[42] S. Buchegger and J.Y. Le Boudec. Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes—Fairness In Dynamic Ad-hoc NeTworks. *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, pages 226–236, 2002.

[43] MT. Refaei, V. Srivastava, L. DaSilva, and M. Eltoweissy. A reputation-based mechanism for isolating selfish nodes in ad hoc networks. *Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. The Second Annual International Conference on*, pages 3–11, 2005.

[44] Q. He, D. Wu, and P. Khosla. SORI: A Secure and Objective Reputation-based Incentive Scheme for Ad-hoc Networks. *IEEE Wireless Communications and Networking Conference (WCNC 2004), Atlanta, GA, USA*, 2004.

[45] L. Buttyán and J.P. Hubaux. Enforcing service availability in mobile ad-hoc WANs. *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, pages 87–96, 2000.

[46] L. Buttyán and J.P. Hubaux. Stimulating Cooperation in Self-Organizing Mobile Ad Hoc Networks. *Mobile Networks and Applications*, 8(5):579–592, 2003.

[47] S. Zhong, J. Chen, and YR Yang. Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks. *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies IEEE*, 3, 2003.

[48] B. Raghavan and A.C. Snoeren. Priority Forwarding in Ad Hoc Networks with Self-interested Parties. *Workshop on Economics of Peer-to-Peer Systems*, 2003.

[49] V. Srivastava, J. Neel, A.B. Mackenzie, R. Menon, L.A. Dasilva, J.E. Hicks, J.H. Reed, and R.P. Gilles. Using game theory to analyze wireless ad hoc networks. *Communications Surveys & Tutorials, IEEE*, 7(4):46–56, 2005.

[50] K. Nakano, R.K Panta, M. Sengoku, and S. Shinoda. On performance of a charging/rewarding scheme in mobile ad-hoc networks. *Circuits and Systems (ISCAS'2005). IEEE International Symposium on*, pages 2962–2966, 2005.

[51] B. Ishibashi and R. Boutaba. Topology and mobility considerations in mobile ad hoc networks. *Ad Hoc Networks*, 3:762–776, 2005.

[52] NS-2. The Network Simulator - ns-2. *www.isi.edu/nsnam/ns/*, 2006.

[53] K. Fall, K. Varadhan, et al. The ns Manual. *The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, Abril*, 2002.

[54] D.B. Johnson, J. Broch, Y.C. Hu, J. Jetcheva, and D.A. Maltz. The CMU Monarch Project's Wireless and Mobility Extensions to ns. *Proc. of 42nd Internet Engineering Task Force*, 1998.

[55] A. Al Hanbali, E. Altman, and P. Nain. A survey of TCP over ad hoc networks. *Communications Surveys & Tutorials, IEEE*, pages 22–36, 2005.

[56] R. Jain. *The art of computer systems performance analysis*. Wiley, 1991.

[57] C. de Waal and M. Gerharz. Bonnmotion: A mobility scenario generation and analysis tool. *Communication Systems group, Institute of Computer Science IV, University of Bonn, Germany. Website: http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion*, 2003.

[58] Y. Liu and YR. Yang. Reputation Propagation and Agreement in Mobile Ad-Hoc Networks. *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, 3, 2003.

[59] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: A Truthful and Cost-efficient Routing Protocol for Mobile Ad Hoc Networks With Selfish Agents. *Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 245–259, 2003.

[60] E. Efstathiou, P. Frangoudis, and G. Polyzos. Stimulating Participation in Wireless Community Networks. In *IEEE INFOCOM 2006*, pages 234–244. 2006.

[61] M. Felegyhazi, L. Buttyan, and J.P. Hubaux. Equilibrium Analysis of Packet Forwarding Strategies in Wireless Ad Hoc Networks–the Static Case. *Personal Wireless Communication (PWC'03)*, page 23—25, 2006.

[62] S. Zhong, L.E. Li, Y.G. Liu, and Y.R. Yang. On Designing Incentive-compatible Routing and Forwarding Protocols in Wireless Ad-Hoc Networks: an Integrated Approach Using Game Theoretical and Cryptographic Techniques. *Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 117–131, 2005.

[63] O. Ileri, S.C. Mau, and N.B. Mandayam. Pricing for Enabling Forwarding in Self-configuring Ad Hoc Networks. *Selected Areas in Communications, IEEE Journal on*, 23(1):151–162, 2005.

# Livros Grátis

( http://www.livrosgratis.com.br )

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo