

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**ROTEAMENTO UTILIZANDO POTENCIAL DE ENTREGA
EM REDES TOLERANTES A ATRASOS**

CRISTINA MOREIRA NUNES

Tese apresentada como requisito parcial à obtenção do grau de Doutor em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul.

Orientador: Prof. Dr. Fernando Luís Dotti

**Porto Alegre
2010**

Dados Internacionais de Catalogação na Publicação (CIP)

N972r Nunes, Cristina Moreira
Roteamento utilizando potencial de entrega em redes tolerantes a atrasos / Cristina Moreira Nunes. – Porto Alegre, 2010.
149 f.

Tese (Doutorado) – Fac. de Informática, PUCRS.
Orientador: Prof. Dr. Fernando Luís Dotti.

1. Informática. 2. Redes Tolerantes a Atrasos.
3. Roteamento – Redes de Computadores. 4. Protocolos de Aplicação sem Fio (Protocolos de Rede de Computadores).
I. Dotti, Fernando Luís. II. Título.

CDD 004.62

**Ficha Catalográfica elaborada pelo
Setor de Tratamento da Informação da BC-PUCRS**



Pontifícia Universidade Católica do Rio Grande do Sul
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

TERMO DE APRESENTAÇÃO DE TESE DE DOUTORADO

Tese intitulada "Roteamento utilizando Potencial de Entrega em Redes Tolerantes a Atrasos", apresentada por Cristina Moreira Nunes, como parte dos requisitos para obtenção do grau de Doutor em Ciência da Computação, Processamento Paralelo e Distribuído, aprovada em 08/04/10 pela Comissão Examinadora:

Prof. Dr. Fernando Luís Dotti -
Orientador

PPGCC/PUCRS

Prof. Dr. Paulo Henrique Lemelle Fernandes -

PPGCC/PUCRS

Prof. Dr. Luciano Paschoal Gasparry -

UFRGS

Prof. Dr. Elias Procópio Duarte Júnior -

UFPR

Homologada em 17/05/2010, conforme Ata No. 18/10 pela Comissão Coordenadora.

Prof. Dr. Fernando Gehm Moraes
Coordenador.

PUCRS

Campus Central

Av. Ipiranga, 6681 - P. 32 - sala 507 - CEP: 90619-900

Fone: (51) 3320-3611 - Fax (51) 3320-3621

E-mail: ppgcc@pucrs.br

www.pucrs.br/facin/pos

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer à Deus pela vida maravilhosa que tem me proporcionado.

Agradeço aos meus pais e minhas irmãs pelo amor, carinho, compreensão e incentivo em todos os momentos. Vocês são muito importantes na minha vida e só tenho a agradecer pelo que fizeram e fazem por mim.

Agradeço, de modo especial, ao Sandro, meu companheiro nesses últimos três anos de doutorado. Agradeço pelo seu amor, paciência, companheirismo e apoio nos momentos mais difíceis. Sem dúvida seu incentivo constante me deu força e coragem para continuar.

Agradeço a minha afilhada Laís pelo seu carinho e pela alegria que dá a minha vida. A toda a minha família pelo carinho e apoio, não só durante a realização deste trabalho, mas na vida inteira.

Agradeço ao meu orientador, Prof. Fernando Luís Dotti, pelos seus ensinamentos e pela sua orientação durante esses longos anos de doutorado. Obrigada por acreditar no meu trabalho.

Agradeço a todos os meus amigos e colegas de trabalho, tanto da PUC quanto do Unilasalle, que sempre me ajudaram e proporcionaram uma excelente convivência ao longo desses anos. Em especial a minha amiga Ana Paula, pelo seu carinho e pelo exemplo de vida que dá a todos.

Agradeço ao Prof. Paulo Fernandes, pelo seu incentivo e pela sua orientação entre os anos de 2008 e 2009. Agradeço, também, ao Prof. João Batista, pelo seu conhecimento, apoio e pelas contribuições para a melhoria deste trabalho.

A todos os meus amigos, não vou listar todos para não esquecer de ninguém, agradeço pela compreensão por minha ausência ao longo desse período.

Por fim, agradeço à PUCRS, em especial à Faculdade de Informática, e ao Unilasalle pelo apoio e confiança a mim depositados.

ROTEAMENTO UTILIZANDO POTENCIAL DE ENTREGA EM REDES TOLERANTES A ATRASOS

RESUMO

Redes Tolerantes a Atrasos (*Delay Tolerant Network* - DTN) são redes sujeitas a interrupções frequentes, altos atrasos e elevadas taxas de erro. Nessas redes os nodos são móveis e fazem o papel de roteadores, armazenando e fazendo o repasse das mensagens que precisam ser entregues. O foco deste trabalho está nas redes DTN estocásticas, nas quais as mudanças topológicas não são conhecidas e surgem técnicas para tentar aumentar a probabilidade de sucesso no roteamento, utilizando conhecimento histórico do comportamento da rede na tomada da decisão. As redes DTN possuem algumas semelhanças com as redes *ad hoc*, mas devido às constantes disrupções que ocorrem nas redes DTN, o problema do roteamento é tratado de forma diferente. Em cenários esparsos, devido à falta de conectividade dos nodos, as redes DTN apresentam maior percentual de mensagens entregues, a custo de maiores atrasos. Um estudo comparando esses dois tipos de redes é realizado por este trabalho, comprovando esse aspecto.

Este trabalho propõe três variações de um protocolo de roteamento para redes DTN estocásticas que utiliza uma métrica chamada de “potencial de entrega” usada na decisão do repasse das mensagens. O potencial de entrega é uma medida histórica que indica o número de encontros em que os nodos estão envolvidos em um período de tempo. No intuito de aumentar a probabilidade de encontrar o destinatário, a estratégia prioriza o roteamento a nodos que tenham um número alto de encontros. O protocolo inicial é chamado de APRP (*Adaptive Potential Routing Protocol*) e, em comparação com outros protocolos para o mesmo tipo de rede, apresenta um percentual de mensagens entregues equivalente a esses protocolos, com um *overhead* bastante reduzido, especialmente para redes esparsas (importantes em DTN). Uma variação do APRP utiliza um mecanismo de confirmação de mensagens já entregues com o intuito de liberar espaço em *buffer* para mensagens novas. Tal variação é chamada de APRP-Ack e apresenta desempenho superior ao APRP, ao custo de um aumento no atraso médio na entrega das mensagens. Por fim, é proposto o APRP-Group, com o objetivo de reduzir o atraso gerado pelo APRP e APRP-Ack. O APRP-Group busca identificar grupos de nodos com potenciais de entrega similares e decidir sobre o repasse das mensagens com base em tais grupos. Este protocolo apresenta uma redução significativa no atraso de entrega, se comparado ao APRP e APRP-Ack, mantendo baixos os custos de comunicação e mantendo ou melhorando níveis de entrega. Os diferentes protocolos foram simulados em um ambiente específico para DTN e comparados com protocolos da literatura, especialmente com o *Spray and Wait* devido ao seu alto desempenho reportado na literatura.

Palavras-chave: Redes Tolerantes a Atrasos; Protocolos de Roteamento.

ROUTING USING DELIVERY POTENTIAL IN DELAY TOLERANT NETWORKS

ABSTRACT

Delay Tolerant Network (DTN) are networks in which frequent interruptions, high delay, and high error rates may occur. In these networks the nodes are mobile and play the role of routers, storing and transferring messages that need to be delivered. This thesis consider the stochastic DTN, where the topological changes are not known in advance and techniques are proposed that try to increase the probability of success in the routing, using, for instance, historical knowledge of network behavior in the routing decision. DTN has some similarities to ad hoc networks, but due to frequent disruptions that occur in DTN, the routing problem is handled differently. In sparse scenarios, due to lack of connectivity of nodes, DTN has higher percentage of delivered messages, at the price of higher delays. A study comparing these two kinds of networks is carried out by this thesis, providing quantitative information about this aspect.

This work proposes three variations of a routing protocol for stochastic DTN based on a metric called "delivery potential", which is used while deciding how messages are forwarded. The delivery potential is a historical measure that indicates the number of meetings in which the nodes are involved in a period of time. In order to increase the probability to find the destination, the strategy gives priority to routing nodes that have a high number of meetings. The initial protocol is called ARPR (Adaptive Potential Routing Protocol) and, in comparison with other protocols for the same type of network, presents a percentage of delivered messages equivalent to these protocols, with a very low overhead, especially for sparse networks (important in DTN). An APRP variation uses an acknowledgement mechanism for messages already delivered in order to release space in buffer for new messages. This variation is called APRP-Ack and has better performance than ARPR, despite causing an increase in the average delay in the delivery of messages. Finally, the APRP-Group is proposed, in order to reduce the delay generated by APRP-Ack. The APRP-Group seeks to identify groups of nodes with similar delivery potential and decide on the forwarding of messages based on these groups. This protocol provides a significant reduction in the delivery delay, compared to APRP and APRP-Ack, maintaining low costs of communication and maintaining or improving levels of delivery. The above mentioned proposed protocols where evaluated using a DTN specific simulation environment specific and compared to analogous protocols from the literature, specially with the Spray and Wait due to its high performance reported in the literature.

Keywords: Delay Tolerant Network; Routing Protocol.

LISTA DE FIGURAS

Figura 1.1	Repasse de uma mensagem feita pelo mecanismo <i>store-carry-forward</i> [Lin03b].	30
Figura 3.1	Atraso na entrega das mensagens para redes DTN. O eixo vertical representa a quantidade de mensagens entregues e o horizontal representa o atraso em segundos.	62
Figura 3.2	Atraso na entrega das mensagens para redes <i>ad hoc</i> . O eixo vertical representa a quantidade de mensagens entregues e o horizontal representa o atraso em segundos.	63
Figura 3.3	Percentual de mensagens entregues cumulativo para redes DTN. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.	64
Figura 3.4	Percentual de mensagens entregues cumulativo para redes <i>ad hoc</i> . O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.	65
Figura 3.5	Percentual de mensagens entregues cumulativo para redes DTN em 10000 segundos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.	66
Figura 4.1	Percentual de mensagens entregues com <i>Random Waypoint</i> . O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.4 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	76
Figura 4.2	Percentual de mensagens entregues com <i>Shortest Path Map Based Movement</i> . O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.5 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	77
Figura 4.3	Percentual de mensagens entregues com <i>Map Based Movement</i> . O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.6 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	77
Figura 4.4	<i>Overhead</i> gerado na rede com <i>Random Waypoint</i> . O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.7 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	78

Figura 4.5	<i>Overhead</i> gerado na rede com <i>Shortest Path Map Based Movement</i> . O eixo vertical representa o número de cópias e o horizontal representa a quantidade de nodos. A Tabela 4.8 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	78
Figura 4.6	<i>Overhead</i> gerado na rede com <i>Map Based Movement</i> . O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.9 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	79
Figura 4.7	<i>Overhead</i> dos protocolos APRP e <i>Spray and Wait</i> com <i>Random Waypoint</i> . O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.7 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	80
Figura 4.8	<i>Overhead</i> dos protocolos APRP e <i>Spray and Wait</i> com <i>Shortest Path Map Based Movement</i> . O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.8 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	80
Figura 4.9	<i>Overhead</i> dos protocolos APRP e <i>Spray and Wait</i> com <i>Map Based Movement</i> . O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.9 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	80
Figura 4.10	Atraso na entrega das mensagens com <i>Random Waypoint</i> . O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.10 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	81
Figura 4.11	Atraso na entrega das mensagens com <i>Shortest Path Map Based Movement</i> . O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.11 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	81
Figura 4.12	Atraso na entrega das mensagens com <i>Map Based Movement</i> . O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.12 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.	81

Figura 4.13	Percentual de mensagens entregues. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.6 apresenta os valores numéricos dessa figura.	82
Figura 4.14	<i>Overhead</i> gerado na rede. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.9 apresenta os valores numéricos dessa figura.	83
Figura 4.15	Atraso na entrega das mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.12 apresenta os valores numéricos dessa figura.	83
Figura 4.16	Mensagens recebidas por grupos de nodos com delta de 0,2. A Tabela 4.13 apresenta os valores numéricos dessa figura.	84
Figura 4.17	Mensagens recebidas por grupos de nodos com delta de 0,5. A Tabela 4.13 apresenta os valores numéricos dessa figura.	84
Figura 5.1	Percentual de mensagens entregues para 5000 mensagens. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 5.1 apresenta os valores numéricos dessa figura.	94
Figura 5.2	Percentual de mensagens entregues para 10000 mensagens. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 5.2 apresenta os valores numéricos dessa figura.	94
Figura 5.3	<i>Overhead</i> gerado na rede para 5000 mensagens. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 5.3 apresenta os valores numéricos dessa figura.	95
Figura 5.4	<i>Overhead</i> gerado na rede para 10000 mensagens. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 5.4 apresenta os valores numéricos dessa figura.	95
Figura 5.5	Atraso na entrega das mensagens para 5000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.5 apresenta os valores numéricos dessa figura.	96
Figura 5.6	Atraso na entrega das mensagens para 10000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.6 apresenta os valores numéricos dessa figura.	96
Figura 5.7	Ocupação de <i>buffer</i> para 5000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.7 apresenta os valores numéricos dessa figura.	96
Figura 5.8	Ocupação de <i>buffer</i> para 10000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.8 apresenta os valores numéricos dessa figura.	97

Figura 5.9	Mensagens descartadas devido a <i>buffer overflow</i> para 5000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.9 apresenta os valores numéricos dessa figura.	97
Figura 5.10	Mensagens descartadas devido a <i>buffer overflow</i> para 10000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.10 apresenta os valores numéricos dessa figura.	97
Figura 5.11	Entregas redundantes para 5000 mensagens. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.11 apresenta os valores numéricos dessa figura.	98
Figura 5.12	Entregas redundantes para 10000 mensagens. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.12 apresenta os valores numéricos dessa figura.	98
Figura 5.13	Tempo de vida das mensagens para 5000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.13 apresenta os valores numéricos dessa figura.	99
Figura 5.14	Tempo de vida das mensagens para 10000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.14 apresenta os valores numéricos dessa figura.	99
Figura 6.1	Distribuição de potenciais para dois tipos de nodos. a) Resultado da simulação com intervalo de velocidades aproximado. b) Resultado da simulação com maior diferença no intervalo de velocidades.	107
Figura 6.2	Distribuição hipotética de potenciais.	109
Figura 6.3	Valores dos desvios padrão se for efetuada a divisão em $k = 1, 2, 3, 4$ e 5 grupos.	111
Figura 6.4	Atraso na entrega das mensagens para 2 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.2 apresenta os valores numéricos dessa figura.	117
Figura 6.5	Atraso na entrega das mensagens para 3 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.3 apresenta os valores numéricos dessa figura.	118
Figura 6.6	Percentual de mensagens entregues para 2 grupos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 6.4 apresenta os valores numéricos dessa figura.	118
Figura 6.7	Percentual de mensagens entregues para 3 grupos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 6.5 apresenta os valores numéricos dessa figura.	118

Figura 6.8	<i>Overhead</i> gerado na rede para 2 grupos. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 6.6 apresenta os valores numéricos dessa figura.	119
Figura 6.9	<i>Overhead</i> gerado na rede para 3 grupos. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 6.7 apresenta os valores numéricos dessa figura.	120
Figura 6.10	Ocupação de <i>buffer</i> para 2 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.8 apresenta os valores numéricos dessa figura.	120
Figura 6.11	Ocupação de <i>buffer</i> para 3 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.9 apresenta os valores numéricos dessa figura.	120
Figura 6.12	Mensagens descartadas devido a <i>buffer overflow</i> para 2 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.10 apresenta os valores numéricos dessa figura.	121
Figura 6.13	Mensagens descartadas devido a <i>buffer overflow</i> para 3 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.11 apresenta os valores numéricos dessa figura.	121
Figura 6.14	Entregas redundantes para 2 grupos. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.12 apresenta os valores numéricos dessa figura.	122
Figura 6.15	Entregas redundantes para 3 grupos. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.13 apresenta os valores numéricos dessa figura.	122
Figura 6.16	Tempo de vida das mensagens para 2 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.14 apresenta os valores numéricos dessa figura.	122
Figura 6.17	Tempo de vida das mensagens para 3 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.15 apresenta os valores numéricos dessa figura.	123
Figura A.1	Modelo SAN do <i>Random Waypoint</i> em uma dimensão.	144
Figura A.2	Distribuição de probabilidade do nodo com variação do tempo de pausa. . .	145
Figura A.3	Modelo SAN do <i>Random Waypoint</i> em duas dimensões.	145
Figura A.4	Distribuição espacial do nodo com variação do tempo de pausa e velocidade de 32 m/s.	146
Figura A.5	Distribuição espacial do nodo em três dimensões.	147
Figura A.6	Random Waypoint na superfície regular e irregular - distribuição espacial do nodo.	147
Figura A.7	Corte feito na metade do eixo <i>y</i> da superfície do triângulo - variando a pausa.	148

LISTA DE TABELAS

Tabela 2.1	Resumo e comparação entre os protocolos de roteamento.	56
Tabela 4.1	Valor médio do potencial para raio de 10 m.	70
Tabela 4.2	Valor médio do potencial para raio de 30 m.	70
Tabela 4.3	Valor médio do potencial para raio de 80 m.	70
Tabela 4.4	Valores do percentual de mensagens entregues com <i>Random Waypoint</i>	85
Tabela 4.5	Valores do percentual de mensagens entregues com <i>Shortest Path Map Based Movement</i>	85
Tabela 4.6	Valores do percentual de mensagens entregues com <i>Map Based Movement</i> . .	86
Tabela 4.7	<i>Overhead</i> gerado na rede com <i>Random Waypoint</i>	86
Tabela 4.8	<i>Overhead</i> gerado na rede com <i>Shortest Path Map Based Movement</i>	87
Tabela 4.9	<i>Overhead</i> gerado na rede com <i>Map Based Movement</i>	87
Tabela 4.10	Atraso na entrega das mensagens com <i>Random Waypoint</i>	88
Tabela 4.11	Atraso na entrega das mensagens com <i>Shortest Path Map Based Movement</i> . .	88
Tabela 4.12	Atraso na entrega das mensagens com <i>Map Based Movement</i>	89
Tabela 4.13	Número médio de mensagens recebidas por grupos de nodos.	89
Tabela 5.1	Valores do percentual de mensagens entregues para 5000 mensagens.	99
Tabela 5.2	Valores do percentual de mensagens entregues para 10000 mensagens. . . .	100
Tabela 5.3	<i>Overhead</i> gerado na rede para 5000 mensagens.	100
Tabela 5.4	<i>Overhead</i> gerado na rede para 10000 mensagens.	101
Tabela 5.5	Atraso na entrega das mensagens para 5000 mensagens.	101
Tabela 5.6	Atraso na entrega das mensagens para 10000 mensagens.	102
Tabela 5.7	Ocupação de <i>buffer</i> para 5000 mensagens.	102
Tabela 5.8	Ocupação de <i>buffer</i> para 10000 mensagens.	103
Tabela 5.9	Mensagens descartadas devido a <i>buffer overflow</i> para 5000 mensagens. . . .	103
Tabela 5.10	Mensagens descartadas devido a <i>buffer overflow</i> para 10000 mensagens. . .	104
Tabela 5.11	Entregas redundantes para 5000 mensagens.	104
Tabela 5.12	Entregas redundantes para 10000 mensagens.	105
Tabela 5.13	Tempo de vida das mensagens para 5000 mensagens.	105
Tabela 5.14	Tempo de vida das mensagens para 10000 mensagens.	106
Tabela 6.1	Exemplo de <i>break_list</i> para k igual a 5.	110
Tabela 6.2	Atraso na entrega das mensagens para 2 grupos.	123
Tabela 6.3	Atraso na entrega das mensagens para 3 grupos.	124
Tabela 6.4	Percentual de mensagens entregues para 2 grupos.	124
Tabela 6.5	Percentual de mensagens entregues para 3 grupos.	125
Tabela 6.6	<i>Overhead</i> gerado na rede para 2 grupos.	125
Tabela 6.7	<i>Overhead</i> gerado na rede para 3 grupos.	126

Tabela 6.8	Ocupação de <i>buffer</i> para 2 grupos.	126
Tabela 6.9	Ocupação de <i>buffer</i> para 3 grupos.	127
Tabela 6.10	Mensagens descartadas devido a <i>buffer overflow</i> para 2 grupos.	127
Tabela 6.11	Mensagens descartadas devido a <i>buffer overflow</i> para 3 grupos.	128
Tabela 6.12	Entregas redundantes para 2 grupos.	128
Tabela 6.13	Entregas redundantes para 3 grupos.	129
Tabela 6.14	Tempo de vida das mensagens para 2 grupos.	129
Tabela 6.15	Tempo de vida das mensagens para 3 grupos.	130

LISTA DE ALGORITMOS

Algoritmo 4.1	Mecanismo de roteamento do APRP.	73
Algoritmo 5.1	Mecanismo de roteamento do APRP-Ack.	92
Algoritmo 6.1	Algoritmo utilizado no agrupamento dos potenciais, baseado em [Ski97]. . .	109
Algoritmo 6.2	Mecanismo de roteamento do APRP-Group.	114
Algoritmo 6.3	Mecanismo de roteamento do APRP-Group - descrição dos procedimentos. .	115

LISTA DE SIGLAS

ADSL	<i>Asymmetric Digital Subscriber Line</i>
AODV	<i>Ad hoc On-demand Distance Vector</i>
APRP	<i>Adaptive Potential Routing Protocol</i>
CAR	<i>Context-aware Adaptive Routing</i>
CBR	<i>Constant Bit Rate</i>
CHANTS	<i>CHallenged NeTworkS</i>
CONDOR	<i>Command and Control On-the-move Network Digital Over-the-horizon Relay</i>
DA-SW	<i>Density-Aware Spray-and-Wait</i>
DSDV	<i>Destination-Sequenced Distance Vector</i>
DSL	<i>Digital Subscriber Line</i>
DSR	<i>Dynamic Source Routing Protocol</i>
DTMNs	<i>Delay Tolerant Mobile Networks</i>
DTN	<i>Delay Tolerant Network</i>
DTNRG	<i>Delay Tolerant Networking Research Group</i>
ED	<i>Earliest Delivery</i>
EPL	<i>Effective Path Length</i>
FIFO	<i>First In First Out</i>
GPS	<i>Global Positioning System</i>
HYMAD	<i>Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks</i>
IETF	<i>Internet Engineering Task Force</i>
IP	<i>Internet Protocol</i>
IPN	<i>Interplanetary Internet</i>
IRTF	<i>Internet Research Task Force</i>
MAC	<i>Medium Access Control</i>
MED	<i>Minimum Expected Delay</i>
MEED	<i>Minimum Estimated Expected Delay</i>
MV	<i>Meeting Visit</i>
NECTAR	<i>Neighborhood Contact History</i>
NS	<i>Network Simulator</i>
O-SW	<i>Oracle-based Spray-and-Wait</i>

PDA	<i>Personal Digital Assistant</i>
PRoPHET	<i>Probabilistic Routing Protocol using History of Encounters and Transitivity</i>
RAPID	<i>Resource Allocation Protocol for Intentional DTN</i>
SAN	<i>Stochastic Automata Networks</i>
SEPR	<i>Shortest Expected Path Routing</i>
SMM	<i>Statistical Mobility Model</i>
TCP	<i>Transmission Control Protocol</i>
TIER	<i>Technology and Infrastructure for Emerging Regions</i>
TTL	<i>Time to Live</i>
UDP	<i>User Datagram Protocol</i>

SUMÁRIO

1. Introdução	29
2. Redes Tolerantes a Atrasos	35
2.1 Aplicações das Redes Tolerantes a Atrasos	36
2.2 Tipos de Contatos	38
2.3 Modelos de Mobilidade	39
2.4 Roteamento em Redes Tolerantes a Atrasos	43
2.4.1 Transmissão Direta	44
2.4.2 Epidêmico	44
2.4.3 Spray and Wait	45
2.4.4 Density-Aware Spray-and-Wait	47
2.4.5 Shortest Expected Path Routing - SEPR	47
2.4.6 Probabilistic Routing Protocol using History of Encounters and Transitivity - PProPHET	48
2.4.7 Meeting Visit - MV	49
2.4.8 Context-aware Adaptive Routing - CAR	49
2.4.9 Minimum Estimated Expected Delay - MEED	50
2.4.10 MaxProp	51
2.4.11 Neighborhood Contact History - NECTAR	51
2.4.12 Core-aided	53
2.4.13 Resource Allocation Protocol for Intentional DTN - RAPID	53
2.4.14 Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks - HYMAD	54
2.4.15 Comparação entre os Protocolos de Roteamento	54
2.5 Métricas de Avaliação para Redes Tolerantes a Atrasos	58
3. Comportamento de Entrega e Atraso em Redes DTN e Ad Hoc	59
3.1 Parametrização dos Experimentos	60
3.2 Avaliação Experimental	61
3.3 Considerações Finais	67
4. O Protocolo APRP	69
4.1 Cálculo do Potencial de Entrega de um Nodo	69
4.2 Roteamento no APRP	71

4.3	Algoritmo	72
4.4	Avaliação Experimental	74
4.4.1	Parâmetros da Simulação	74
4.4.2	Percentual de Mensagens Entregues	76
4.4.3	Overhead	78
4.4.4	Atraso na Entrega das Mensagens	80
4.4.5	Análise do Delta	82
4.4.6	Avaliação da Sobrecarga nos Nodos	83
4.4.7	Tabelas com Valores dos Gráficos	84
4.5	Considerações Finais	90
5.	O Protocolo APRP-ACK	91
5.1	Algoritmo	91
5.2	Avaliação Experimental	93
5.2.1	Percentual de Mensagens Entregues	93
5.2.2	Overhead	94
5.2.3	Atraso na Entrega das Mensagens	95
5.2.4	Ocupação de Buffer	96
5.2.5	Mensagens Descartadas Devido a Buffer Overflow	97
5.2.6	Entregas Redundantes	98
5.2.7	Tempo de Vida das Mensagens	98
5.2.8	Tabelas com Valores dos Gráficos	99
5.3	Considerações Finais	106
6.	O Protocolo APRP-Group	107
6.1	Caracterização e Agrupamento de Potenciais	107
6.2	Roteamento no APRP-Group	111
6.3	Algoritmo	112
6.4	Avaliação Experimental	116
6.4.1	Atraso na Entrega das Mensagens	117
6.4.2	Percentual de Mensagens Entregues	118
6.4.3	Overhead	119
6.4.4	Ocupação de Buffer	120
6.4.5	Mensagens Descartadas devido a Buffer Overflow	121
6.4.6	Entregas Redundantes	121
6.4.7	Tempo de Vida das Mensagens	122
6.4.8	Tabelas com Valores dos Gráficos	123
6.5	Considerações Finais	130

7. Conclusão	131
Bibliografia	135
A. Modelagem do Random Waypoint	143
A.1 Modelagem de Superfícies Quadradas em SAN	144
A.2 Modelagem de Superfícies Irregulares em SAN	147

1. Introdução

Redes Tolerantes a Atrasos (*Delay Tolerant Network* - DTN) são redes sem fio conectadas ocasionalmente que podem sofrer de particionamentos frequentes e podem compreender uma ou mais famílias de protocolos [Cer07]. Assim como as redes *ad hoc*, as redes DTN são formadas por múltiplos nodos que se comunicam sem a necessidade de estações base ou de uma rede com infra-estrutura fixa pré-existente. Contudo, segundo [Jai04], as desconexões nas redes DTN são frequentes, muitas vezes não existindo um caminho entre uma origem e um destino que desejem se comunicar. As principais razões mencionadas na literatura para as desconexões são a mobilidade dos nodos e a operação em ambientes hostis [Zha06].

Em redes onde há movimentação dos nodos, a comunicação direta entre um par de nodos em um determinado instante somente é possível quando a distância entre eles é menor do que o alcance da transmissão. Isso significa que, mesmo que as mensagens sejam roteadas através de nodos que fazem o papel de roteadores, não há garantia de que se encontre um caminho de comunicação viável até o destino. Já em ambientes hostis, as desconexões acontecem devido à interferência no sinal, impossibilitando a comunicação [Zha06].

A falta de conectividade entre os nodos também é considerada em redes *ad hoc*, entretanto a abordagem utilizada é diferente. Para resolver o roteamento em redes *ad hoc*, supõe-se a possibilidade de um caminho contemporâneo válido entre origem e destino. Diversos protocolos foram especificados para tais redes, como: AODV (*Ad hoc On-demand Distance Vector*), DSR (*Dynamic Source Routing Protocol*) e DSDV (*Destination-Sequenced Distance Vector*) [Liu05]. Tais protocolos são variações de protocolos de roteamento para redes tradicionais e utilizam o mecanismo *store-and-forward* no repasse das mensagens. Por outro lado, nas redes DTN esse mecanismo não funciona adequadamente, dada a inexistência de caminhos contemporâneos de comunicação entre os nodos.

Para trabalhar com conectividade intermitente, uma abordagem utilizada é estender o roteamento *store-and-forward* para o roteamento *store-carry-forward* [Zha06]. Neste mecanismo, um nodo pode armazenar uma mensagem até encontrar outro nodo propício para o roteamento em direção ao destinatário. A decisão de para qual(is) nodo(s) repassar uma mensagem, de forma a aumentar sua probabilidade de alcançar o destino, é uma das principais questões no projeto de um protocolo de roteamento para redes DTN. Esse repasse também pode ser feito para um único ou para vários nodos. Segundo [Abd07], a abordagem de única cópia tenta reduzir o uso de *buffer* e o número de mensagens transferidas, mas sofre de grandes atrasos e baixa taxa de entrega. Esquemas de múltiplas cópias, por outro lado, alcançam atrasos mais baixos e maior taxa de entrega.

A Figura 1.1 mostra como a mobilidade dos nodos pode ser usada para eventualmente entregar uma mensagem para seu destino. A figura representa uma sequência de instantes do tempo, bem como a localização dos nodos móveis. Nesta figura, o nodo A tem uma mensagem para ser entregue ao nodo D, mas não há um caminho entre esses dois nodos. Como mostra a Figura 1.1a-d, a

mobilidade dos nodos permite que a mensagem primeiro seja passada para o nodo B, então para o C e finalmente C se move para próximo de D e pode entregar a mensagem para seu destino final.

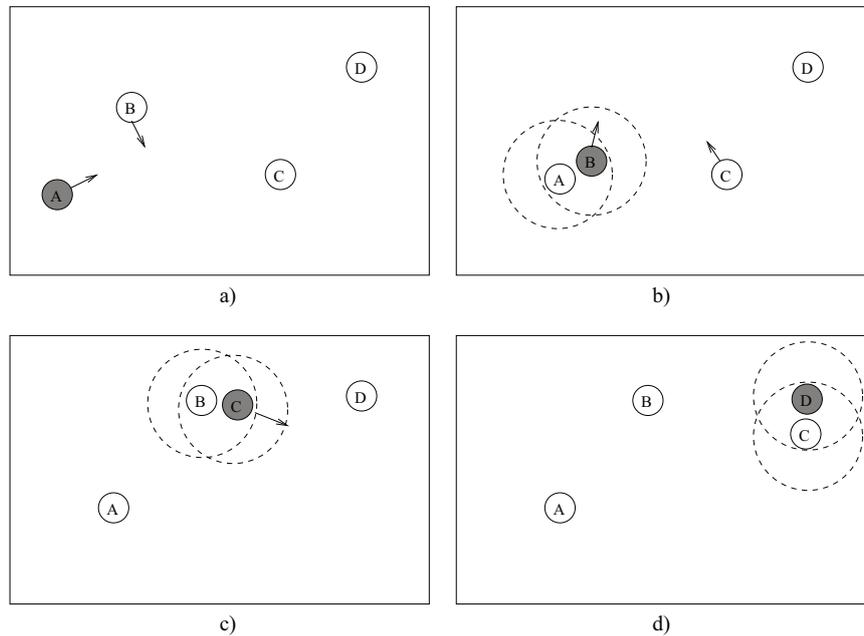


Figura 1.1: Repasse de uma mensagem feita pelo mecanismo *store-carry-forward* [Lin03b].

Segundo [Zha06], as redes DTN podem ser divididas em determinísticas e estocásticas e diferentes protocolos de roteamento são necessários. Quando as mudanças que ocorrem na topologia da rede são determinísticas e conhecidas, ou ao menos previsíveis, o repasse das mensagens pode ser escalonado, podendo-se calcular as jornadas das mensagens a priori [BX03]. Já quando as mudanças topológicas não são conhecidas, um repasse aleatório pode ser utilizado. Nesse sentido, surgem técnicas para tentar aumentar a probabilidade de sucesso no roteamento, utilizando conhecimento histórico do comportamento da rede para tomar a decisão.

Essa segunda classe é atrativa pois várias aplicações podem utilizar esse tipo de rede, vide Seção 2.1, sendo necessário que se tenham protocolos de roteamento eficientes para trabalhar nessas redes. Além disso, os protocolos de roteamento para redes DTN estocásticas devem assumir que os nodos não possuem nenhuma informação sobre a rede, bem como sobre os outros nodos da rede.

Além disso, as decisões de para qual ou quais nodos deve-se repassar as mensagens nas redes DTN estocásticas não são simples e a literatura apresenta diversos trabalhos que propõem mecanismos de roteamento para essas redes. Uma estratégia que pode ser empregada é a utilização de *flooding*, distribuindo cópias duplicadas das mensagens para todos os nodos da rede, como em [Vah00], ou para uma parte deles, como em [Bur06], [Bal07] e [Spy08]. Contudo, a maioria dos protocolos propostos baseia a decisão do repasse das mensagens a partir de informações históricas acerca do comportamento dos nodos. Uma métrica bastante utilizada é a probabilidade de entrega das mensagens para o destino, sendo que o repasse ocorre somente para nodos que tenham uma maior probabilidade de encontrar o destino no futuro. Essa métrica foi proposta em [Lin03b], mas vários protocolos também se baseiam nessa medida, como em [Mus05] e [Bur06]. A frequência de encontro

entre pares de nodos e suas visitas em certas regiões geográficas é uma outra métrica utilizada, conforme definido em [Bur05]. Em [Jon05] a métrica utilizada diz respeito ao tempo em que ocorreram os encontros entre os nodos, sendo que o repasse ocorre para nodos que se encontraram mais recentemente. Já [Oli09d] utiliza em conjunto a probabilidade de entrega e o tempo dos contatos. Além disso, em [Abd07] o repasse ocorre para os nodos que possuem mais recursos na rede.

Como pode-se observar, várias métricas diferentes são propostas pelos protocolos apresentados na literatura. A partir dessa observação, pode-se categorizar os protocolos de roteamento de acordo com as informações que são utilizadas no repasse das mensagens. Vários protocolos de roteamento, como [Lin03b], [Mus05], [Bur06], fazem o repasse das mensagens considerando dados históricos e informações como probabilidades de entrega por nodo ou por mensagem. Apesar desses protocolos atingirem uma razoável probabilidade de entrega das mensagens, vide Seção 4.4, essas informações ficam armazenadas nos nodos, podendo acarretar problemas de espaço em memória devido à quantidade de informações armazenadas. Além disso, uma grande quantidade de informações de controle pode ser repassada na rede à medida que o tamanho da mesma aumenta. Por outro lado, alguns protocolos de roteamento, como [Vah00], [Spy08], [Tou09], não consideram informações sobre a rede no repasse das mensagens, repassando mensagens aleatoriamente para qualquer nodo encontrado. Normalmente essas técnicas são restrições do *flooding*, e acredita-se que, apesar do bom percentual de mensagens entregues, vide Seção 4.4, esse percentual poderia ser melhorado caso informações a respeito da rede fossem consideradas no repasse das mensagens.

Identifica-se, desta forma, um nicho de investigação no sentido de elaborar um mecanismo de decisão de repasse baseado em observações mínimas da rede. Tal mecanismo tem um custo de manutenção independente do número de nodos e visa aumentar a probabilidade de entrega, principalmente se comparado a técnica de repasse aleatório, como a proposta em [Spy08].

Neste contexto, este trabalho propõe o uso de uma medida chamada “potencial de entrega” para nortear as decisões de repasse. O potencial de entrega identifica o número de encontros em que os nodos estão envolvidos em um período de tempo e foi proposto em [Nun09a]. Parte-se da hipótese que a conectividade dos nodos influi no sucesso do repasse, ou seja, nodos que historicamente têm mais contato com outros nodos devem ter mais chance de encaminhar a mensagem com sucesso ao destino. Essa medida é importante pois os nodos não precisam ter informações sobre os demais nodos da rede, o que torna a estratégia simples e não sobrecarrega a rede com informações de controle, mesmo com o aumento de sua densidade. Além disso, o protocolo proposto considera que a localização geográfica dos nodos não é conhecida e não há conhecimento da presença do destino na rede. A partir dessa medida, ou observação da rede, três protocolos são apresentados e avaliados neste trabalho, em ordem crescente de complexidade.

O protocolo inicialmente proposto é chamado de APRP (*Adaptive Potential Routing Protocol*) [Nun09a] e prioriza o roteamento a nodos que participem de um maior número de encontros no intuito de aumentar a probabilidade de encontrar o destinatário. A técnica proposta se mostra de simples implementação e com bom potencial de escalabilidade, já que os nodos mantêm um conjunto

bastante reduzido de informações sobre a rede, além de explorar o movimento dos nodos e o contato com seus vizinhos.

Três protocolos de roteamento foram escolhidos na realização de experimentos e comparação de desempenho com o APRP, sendo eles Epidêmico [Vah00], *Probabilistic Routing Protocol using History of Encounters and Transitivity* (PRoPHET) [Lin03b] e *Spray and Wait* [Spy08]. Tais protocolos são bastante referenciados na literatura e apresentam bons níveis de desempenho. Contudo, as comparações são especialmente realizadas com o protocolo *Spray and Wait*, pois o APRP utiliza o mecanismo de controle do número de cópias das mensagens proposto por esse protocolo. De acordo com os experimentos realizados, vide Seção 4.4, observa-se que os resultados obtidos do APRP em comparação com esses protocolos variam de acordo com o cenário utilizado, mas em linhas gerais o APRP gerou uma considerável diminuição de *overhead*¹, principalmente quando comparado aos protocolos PRoPHET e Epidêmico, mantendo níveis de entrega compatíveis com o *Spray and Wait*. Esse comportamento é especialmente observado em redes esparsas, de maior importância no caso de redes DTN, vide Seção 3.2. Como exemplo, considerando as nove configurações mais significantes² para o padrão de mobilidade *Map Based Movement* [Ekm08], o APRP apresentou, quando comparado ao *Spray and Wait*, redução no percentual de mensagens entregues em média de 1,2%, redução no *overhead* em média de 43,2% e aumento no atraso em média de 31,8%. Já em uma configuração específica, com raio de alcance de 80 m, 50 nodos na área e com 5000 mensagens sendo criadas, o protocolo APRP reduziu o *overhead* em 27,8%, aumentou a probabilidade de mensagens entregues em 2,0% e gerou um aumento no atraso médio na entrega das mensagens em 54,9%. Esses resultados foram obtidos em comparação ao protocolo *Spray and Wait*, com padrão de mobilidade *Map Based Movement*.

Como extensão ao APRP, propõe-se o APRP-Ack [Nun09b]. Esse protocolo mantém os objetivos definidos para o APRP e usa mensagens de confirmação para liberar espaço de armazenamento em nodos intermediários. O APRP-Ack utiliza a mesma estratégia de repasse proposta para o APRP. Conforme experimentos realizados, vide Seção 5.2, o APRP-Ack, tem em geral, melhores índices de desempenho que o APRP (aumentando o percentual de mensagens entregues e diminuindo o *overhead*). Quando comparado a outros protocolos referenciados na literatura, como o *Spray and Wait*, o APRP-Ack tem um maior percentual de mensagens entregues, gerando menos cópias de mensagens na rede. Considerando a média das dezoito configurações mais significantes³, conforme apresentado anteriormente, o APRP apresentou, quando comparado ao *Spray and Wait*, aumento no percentual de mensagens entregues em média de 3,0%, redução no *overhead* em média de 51,5% e aumento no atraso em média de 45,4%.

O terceiro protocolo proposto neste trabalho é chamado de APRP-Group [Nun10], o qual estende ambos protocolos, APRP e APRP-Ack. A partir dos resultados apresentados observou-se que o

¹Considera-se *overhead* neste trabalho como sendo o número de cópias geradas na rede para cada mensagem criada.

²Com base nos estudos realizados no Capítulo 3, considerou-se apresentar apenas os resultados dos cenários com menor densidade, pois considera-se que esses são os cenários propícios para o uso de redes DTN.

³Nesse caso foram realizados experimentos com 5000 e 10000 mensagens no intuito de testar a eficiência do Ack.

atraso médio na entrega das mensagens de ambos APRP e APRP-Ack foi um pouco maior que nos demais protocolos. Isso é devido ao repasse não acontecer imediatamente no próximo encontro com outro nodo, mas depender do nodo encontrado ter um valor de potencial de entrega que justifique o repasse realizado. O APRP-Group visa reduzir esse atraso médio gerado pelos protocolos APRP e APRP-Ack, sem prejudicar a taxa de entrega das mensagens. Conforme discutido na Seção 6.1, observa-se que dependendo do padrão de movimentação, existem nodos com valores de potenciais de entrega agrupados. O APRP-Group se utiliza dessa característica, agrupando logicamente nodos com potenciais de entrega semelhantes e definindo uma nova estratégia de repasse baseada nos potenciais desses grupos. Desta forma, ao invés de utilizar o valor de potencial individual de cada nodo para decidir sobre o repasse de uma mensagem, utilizam-se os valores de potenciais dos grupos para realizar o repasse. Contudo, para nodos do grupo de maior potencial, é permitida a troca de mensagens dentro do mesmo grupo. Assim, evita-se que mensagens fiquem retidas durante muito tempo em nodos com alto potencial de entrega. Além disso, com o objetivo de aumentar a disseminação de mensagens na rede, foi definido um novo mecanismo para administração de cópias de uma mensagem. Através desse mecanismo, dedica-se parte das cópias para repasse com baixo atraso, independente do potencial, e outra parte segundo a estratégia do potencial.

Contudo, devido aos mecanismos utilizados pelo APRP para reduzir o atraso na entrega das mensagens, os objetivos definidos para o APRP e APRP-ACK não são totalmente mantidos. O APRP-Group se torna mais complexo que seus predecessores, devido à técnica de agrupamento utilizada. Além disso, ele armazena e troca informações sobre os demais nodos da rede, sobrecarregando mais os nodos com informações de controle. Apesar dessas características, de acordo com experimentos realizados, o APRP-Group apresenta parâmetros como atraso e percentual de mensagens entregues superiores ao APRP-Ack. Já em comparação com o protocolo *Spray and Wait*, o APRP-Group desempenha de forma análoga em redes densas e melhor em redes esparsas, características de redes DTN, vide Seção 3.2. Como exemplo, considerando a média das dezoito configurações mais significantes⁴, conforme apresentado anteriormente, o APRP-Group apresentou, quando comparado ao *Spray and Wait*, aumento no percentual de mensagens entregues em média de 2,3%, redução no *overhead* em média de 13,8% e aumento no atraso em média de 9,3%. Em um cenário específico, onde dois tipos de grupos foram simulados com o mesmo padrão de mobilidade, com raio de alcance de 30 m, com 50 nodos na área e com 5000 mensagens sendo criadas, o protocolo APRP-Group apresentou redução no *overhead* em até 20,7%, aumento na probabilidade de mensagens entregues em até 9,3% e aumento no atraso médio na entrega das mensagens em até 22,3%. Esses resultados foram obtidos em comparação ao protocolo *Spray and Wait*.

A estrutura deste trabalho é apresentada a seguir. O Capítulo 2 apresenta uma descrição mais detalhada sobre redes DTN, discutindo aspectos que interferem no desempenho dessas redes, como modelos de mobilidade por exemplo. Além disso, são apresentados protocolos de roteamento propostos para redes DTN estocásticas, bem como métricas utilizadas para medir o desempenho desses

⁴Nesse caso foram realizados experimentos com diferentes tipos de grupos no intuito de testar a eficiência da técnica de agrupamento proposta.

protocolos. O Capítulo 3 apresenta um estudo do impacto da densidade da rede em relação a percentual de mensagens entregues e atraso, utilizando técnicas *ad hoc* e DTN. Esse estudo também serve para identificar que densidade da rede deve-se investigar técnicas DTN. O Capítulo 4 descreve o protocolo APRP, o seu funcionamento e apresenta resultados obtidos a partir dos experimentos realizados. O Capítulo 5 apresenta o APRP-Ack e discute sobre os resultados obtidos da comparação entre APRP, APRP-Ack e *Spray and Wait*. O Capítulo 6 descreve o APRP-Group, apresentando o mecanismo utilizado para agrupar nodos de acordo com o potencial de entrega, bem como a estratégia de repasse utilizada. Além disso, esse capítulo discute os resultados obtidos a partir de experimentos usando dois e três tipos de nodos. O Capítulo 7 apresenta as conclusões do trabalho. Por fim, o Apêndice A mostra a modelagem de um padrão de mobilidade, bastante empregado em análises de redes sem fio em geral, usando *Stochastic Automata Networks* (SAN) [Fer98].

2. Redes Tolerantes a Atrasos

As redes DTN fornecem comunicação em ambientes altamente desafiadores, com conectividade intermitente, longos e variados atrasos e com alta taxa de erros [Sco07]. Isso contrasta com as redes tradicionais baseadas no protocolo IP (*Internet Protocol*), que possuem características como: conectividade contínua, baixa perda de pacotes e atraso razoavelmente pequeno [Zha06]. Apesar do termo DTN (*Delay Tolerant Network*) ser o mais utilizado na literatura, também podem ser encontradas outras terminologias, tais como: Redes Desafiadoras (*CHALLENGED NeTworkS* (CHANTS)) [War03], Redes Tolerantes a Desconexões (*Disruption Tolerant Networks* (DTN)) [Bal07], Redes Móveis Tolerantes a Atrasos (*Delay Tolerant Mobile Networks* (DTMNs)) [Har05], Redes Móveis Esparsas Desconectadas (*Disconnected Sparse Mobile Networks*) [Har09], entre outras.

Segundo [Cer07] e [Spy08], como exemplos de redes DTN, tem-se: redes interplanetárias, redes de sensores, redes sem fio terrestres em áreas remotas, redes de satélites, redes acústicas embaixo d'água, redes terrestres móveis, redes *ad hoc* militares, redes *ad hoc* veiculares, entre outras. Nesses cenários, o atraso fim-a-fim é bastante alto, podendo ser de vários segundos ou mesmo horas, sendo muito maior do que o atraso suportado por protocolos, como TCP (*Transmission Control Protocol*). Além disso, caminhos fim-a-fim completamente conectados entre duas entidades que desejam se conectar são raros ou nunca existirão.

Nesse sentido, dentro do *Internet Research Task Force* (IRTF)¹ foi criado um grupo de pesquisa para redes tolerantes a atrasos, o *Delay Tolerant Networking Research Group* (DTNRG)², com o objetivo de desenvolver uma arquitetura e tecnologias de suporte para esse tipo de rede. A arquitetura, descrita em [Cer07], é uma evolução da proposta do Projeto da Internet Interplanetária (*Interplanetary Internet* (IPN)) [Bur03] e descreve como um conjunto de nodos se organiza para armazenar e encaminhar mensagens cooperativamente.

Segundo [Fal03], a arquitetura DTN inclui os conceitos de região e *gateways*. Uma região corresponde a um agrupamento de nodos (fixos ou móveis) com características relativamente homogêneas, na qual sua formação e o número de nodos participantes variam de acordo com a natureza do sistema [Oli09a]. Um *gateway* pode ser identificado como qualquer nodo móvel que participe de pelo menos duas regiões. Eles são utilizados como pontos de ligação entre as regiões e servem para traduzir protocolos, bem como para aplicar políticas e controles [Fal03].

Os *gateways* trabalham em uma camada de agregação definida pela arquitetura da DTN, chamada de *Bundle Layer* [Cer07]. Esta camada é uma sobre camada (*overlay*) utilizada para tornar a rede DTN totalmente independente das diversas redes regionais, permitindo que as aplicações se comuniquem através de múltiplas regiões. Essa camada está localizada acima do nível de transporte e abaixo do nível de aplicação, e emprega o armazenamento persistente para ajudar a diminuir o impacto das interrupções que ocorrem na rede. Dessa forma, ao invés dos *gateways* implemen-

¹<http://www.irtf.org/>

²<http://www.dtnrg.org/>

tarem uma estratégia de *store-and-forward* como ocorre nas redes convencionais, uma estratégia chamada de *store-carry-forward* é utilizada [Zha06]. De acordo com [Lin09], nesse tipo de roteamento as mensagens podem ficar armazenadas nos nodos intermediários por um tempo maior do que ocorre nas filas dos roteadores tradicionais. Assim, através da mobilidade dos nodos é possível que eventualmente a mensagem seja entregue ao destino.

Um dos grandes desafios das redes DTN está relacionado ao roteamento das mensagens. Devido à mobilidade dos nodos e a fatores ambientais, pode não haver um caminho contemporâneo válido entre os nodos que desejem se comunicar. Dessa forma, os protocolos de roteamento utilizados em redes tradicionais não funcionam adequadamente nesse tipo de rede e novos protocolos e algoritmos precisam ser desenvolvidos para diferentes cenários.

Este trabalho considera um subconjunto das redes cobertas pela arquitetura DTN, tratando apenas do roteamento entre regiões. Dessa forma, os *gateways* são chamados de nodos da rede DTN.

A seguir estão descritos exemplos de utilização de redes DTN, bem como características que influenciam no funcionamento das mesmas. Além disso, é apresentada uma revisão de protocolos de roteamento para redes DTN estocásticas, bem como métricas utilizadas para avaliar o desempenho desses protocolos.

2.1 Aplicações das Redes Tolerantes a Atrasos

As redes DTN podem ser encontradas em vários tipos de ambientes e para várias aplicações diferentes. A seguir estão descritos alguns projetos que implementam tais redes:

- CONDOR (*Command and Control On-the-move Network Digital Over-the-horizon Relay*) [CON09] [Sco05]: é um projeto da MITRE Corporation para prover comunicação, através de veículos táticos, entre as tropas que estão em campos de batalha e seu quartel general. Três tipos diferentes de veículos foram projetados: *Gateways*, Pontos de Presença e Comando e Controle. Os *Gateways* são equipados com rádios para permitir a cobertura de comunicação em locais de difícil visão. Os Pontos de Presença permitem que unidades individuais se conectem à rede de dados. O Comando e Controle funciona como um posto de comando móvel onde as comunicações de dados ocorrem. O projeto inclui roteadores, equipamento de criptografia, rádio e modem por satélite. Apesar dos veículos realizarem comunicação por satélite, desconexões podem ocorrer. Para aumentar a confiabilidade, uma DTN foi projetada. Desta forma, as mensagens enviadas têm uma maior chance de serem recebidas, pois mesmo que a comunicação com o satélite seja perdida, os dados ficam armazenados e podem ser enviados no momento de um novo estabelecimento da conexão com o mesmo.
- DakNet [Pen04]: é um projeto liderado pela *First Mile Solutions* [Fir09] com o objetivo de levar comunicação digital de baixo custo a aldeias e vilas remotas. DakNet é uma rede sem fio que fornece conectividade digital de forma assíncrona. Veículos como ônibus e motocicletas

são utilizados como pontos de acesso, levando comunicação a áreas rurais à medida que se movimentam. Ao entrar no raio de alcance de uma vila, os pontos de acesso automaticamente recebem e transmitem os dados armazenados de locais chamados quiosques. Ao se mover para um centro maior com acesso à Internet, os pontos de acesso transmitem as informações coletadas e armazenam outras informações recebidas da Internet para posteriormente repassá-las aos quiosques. Esse projeto foi implementado com sucesso em partes remotas da Índia, Camboja, Ruanda e Costa Rica com um custo muito menor que as soluções tradicionais de rede fixa.

- DieselNet [Die09]: é um projeto da Universidade de Massachusetts Amherst. O projeto consiste em uma rede com 40 ônibus cobrindo uma área de aproximadamente 240 km² ao redor de Amherst, MA. Os ônibus ligam-se uns aos outros de forma intermitente e cada ônibus transporta rádios 802.11b e um computador com recursos limitados. O acesso à rede é fornecido a passageiros e transeuntes. Um dispositivo GPS (*Global Positioning System*) registra o horário e as localizações dos ônibus. Para aumentar a conectividade entre os nodos móveis da DieselNet, foram projetados *throwboxes* que funcionam como roteadores estacionários, armazenando e enviando dados entre os ônibus. Os *throwboxes* são colocados em prédios ao longo das rotas dos ônibus e utilizam baterias ou energia solar para funcionar.
- Drive-thru Internet [Ott06]: este projeto fornece serviços da Internet em ambientes com conectividade intermitente, mais especificamente para veículos em movimento. A ideia é fornecer pontos de acesso (chamados de *Drive-thru proxy*) ao longo de uma estrada, dentro de uma cidade, ou até mesmo em estradas de alta velocidade, tais como rodovias. Tais pontos de acesso precisam ser colocados de forma que um veículo em movimento possa obter acesso à rede por algum período de tempo (relativamente curto). Caso o ponto de acesso esteja localizado em áreas de descanso, o motorista pode sair do carro, passar devagar ou mesmo parar para prolongar o período de conectividade. Um ou mais pontos de acesso localmente interligados formam uma ilha de conectividade que pode fornecer serviços locais, bem como acesso à Internet. Várias dessas ilhas de conectividade ao longo de uma estrada ou na mesma área geográfica podem estar interligadas e cooperar para fornecer acesso à rede com conectividade intermitente para uma área maior. O *Drive-Thru proxy* atua como um ponto fixo: ele busca e armazena conteúdo da Internet em nome do nodo móvel e repassa esses dados quando tal nodo móvel está alcançável. Em tal rede, a tecnologia IEEE 802.11 é utilizada.
- Projeto N4C (*Networking for Communications Challenged Communities: Architecture, Test Beds and Innovative Alliances*) [N4C09]: este projeto visa dar acesso de comunicação via banda larga a áreas onde não há cobertura via ADSL (*Asymmetric Digital Subscriber Line*), satélites ou 3G. O projeto N4C visa o desenvolvimento de uma DTN para tais áreas através de dois *test beds*. Um deles estará localizado no norte da Suécia, e o outro na zona das montanhas da Eslovênia. Alguns dos objetivos do projeto são: fazer o monitoramento de renas, monitorar pessoas que caminham nas colinas e obter dados meteorológicos e ambientais.

- Seaweb [Ric05]: é um projeto da *Naval Postgraduate School* de Monterey, Califórnia, que tem o objetivo de conectar nodos fixos e móveis através de uma ampla área (100 a 10000 km²) no ambiente subaquático. Os dispositivos se comunicam através de comunicação acústica, sendo equipados com modems de telesemar. Através dessa rede é possível a localização de sensores e a navegação de nodos móveis, como submarinos e veículos submarinos autônomos. Diversos *gateways* de comunicação servem de interface dos nodos distribuídos com os centros de comando.
- TIER (*Technology and Infrastructure for Emerging Regions*) [TIE09]: é um projeto da Universidade da Califórnia em Berkeley que investiga a aplicação de novas tecnologias para regiões em desenvolvimento. Esse projeto foca na aplicação de uma infra-estrutura de hardware e software projetada para a realidade física, política e econômica de áreas em desenvolvimento. TIER definiu um sistema de arquivos, chamado de *TierStore*, que esconde a complexidade da conectividade intermitente e simplifica a implantação de aplicações como e-mail, Web, entre outros.
- TurtleNet [Tur09]: é um projeto da Universidade de Massachusetts Amherst que visa monitorar o comportamento e o habitat de um tipo de tartaruga que vive em algumas regiões nos Estados Unidos e no Canadá. Pesquisadores têm um interesse particular nessas tartarugas pois seus números estão em rápido declínio devido à destruição de seu habitat, sendo que o monitoramento até então era feito de forma manual. Em agosto de 2008, foram implantados dispositivos com GPS em 13 tartarugas no sul do Mississippi.
- Zebranet [Jua02]: é um projeto da Universidade de Princeton e consiste em examinar o comportamento das zebras em seu habitat. Cada animal carrega uma coleira de rastreamento. As coleiras possuem GPS, memória *flash*, transceptores sem fio e um processador pequeno. Periodicamente a localização das zebras é captada pelo GPS e suas coordenadas são armazenadas na memória *flash*. Como não há serviço de telefonia celular ou de transmissão de comunicação abrangendo a região onde os animais são estudados, as coleiras funcionam como uma rede ponto a ponto, repassando os dados até que os mesmos sejam recebidos pela estação base. Contudo, como a estação base que recebe os dados das coleiras é móvel (pode ser um pesquisador se locomovendo por terra ou pelo ar), pode ocorrer de algumas coleiras não conseguirem enviar seus dados para a estação base.

2.2 Tipos de Contatos

Em redes DTN, não se assume que todos os nodos sejam alcançáveis, podendo ser contatados a qualquer instante, por isso, um conceito importante que é considerado na arquitetura da DTN é o tipo de contato. Este conceito está diretamente ligado à transferência das mensagens, pois quando dois nodos se encontram, uma transferência de mensagens pode ocorrer. De acordo com [Cer07], a

arquitetura DTN classifica os contatos em: persistentes, sob demanda, programados, previsíveis e oportunistas. A descrição de cada um deles está apresentada a seguir.

- **Contato Persistente:** é aquele tipo de contato que está sempre disponível, onde nenhuma inicialização é necessária. Um exemplo desse tipo de contato seria uma conexão via DSL (*Digital Subscriber Line*) ou *cable modem*.
- **Contato sob Demanda:** precisa ser instanciado e posteriormente funciona como um contato persistente até ser encerrado. Como exemplos desse tipo de contato tem-se uma conexão discada (*dial-up*) ou redes de sensores que requerem o envio de uma mensagem específica para “acordar” os sensores que estão “dormindo” [Oli09b].
- **Contato Programado:** presume que a troca de informações possa ser programada previamente com o horário e duração do contato. Um exemplo desse tipo de contato poderia ser a comunicação da Terra com um satélite de baixa órbita. O satélite gira em torno da Terra e pode ser “visto” de um determinado ponto na mesma em períodos definidos. Outro exemplo seria em uma rede de sensores onde determinados nós “acordam” em horários pré-estabelecidos, voltando a “dormir”, para poupar energia, fora dos horários programados [Oli09b]. Este tipo de contato pode ser considerado como um contato intermitente.
- **Contato Previsível:** os nodos fazem previsões sobre o horário e a duração dos contatos com base em dados históricos. Esse tipo de contato possui um certo grau de incerteza, pois os contatos são estimados. Dado um nível de segurança suficiente, as rotas podem ser escolhidas com base nas informações de experiências passadas [Oli09b]. Este tipo de contato também pode ser considerado como um contato intermitente.
- **Contato Oportunista:** ocorre através de encontros não programados entre os nodos. É possível tirar proveito de um contato feito por acaso para realizar uma transferência de dados. Um exemplo pode ser um avião sobrevoando uma área e anunciando sua disponibilidade em receber ou transmitir dados. Já outro exemplo poderia ser a comunicação de um enlace *Bluetooth* ou infra-vermelho de um PDA (*Personal Digital Assistant*) com um quiosque no saguão de um aeroporto. À medida que o PDA se aproxima do quiosque a comunicação pode ocorrer e seu término acontece quando o enlace é perdido ou desligado.

2.3 Modelos de Mobilidade

Um das características de uma rede DTN é a possibilidade de movimentação dos nodos. Devido a essa movimentação, ocorre uma mudança constante na topologia da rede, influenciando a troca de mensagens realizadas pelos nodos e dificultando a sua análise [San02]. A troca da topologia ocasiona alguns problemas como, por exemplo, desconexões frequentes, grandes atrasos e altas taxas de erro [Lin03a]. Portanto, é de extrema importância a forma como os nodos se movimentam na rede. Pesquisas anteriores, como em [Cam02] e [Jar03], mostraram que os padrões de mobilidade

podem influenciar no desempenho dos protocolos usados, alterando a taxa de entrega dos pacotes, por exemplo, como pode ser visto na Seção 4.4.

A análise do movimento dos nodos das redes DTN, assim como das redes sem fio em geral, normalmente é realizada através de estudos analíticos - vide Apêndice A e [Del06] - ou de simulação. Ao se realizar a análise de uma rede, é necessário representar o comportamento móvel dos nodos na mesma. Para esse fim, foram desenvolvidos padrões de mobilidade que tentam reproduzir o comportamento dos nodos de uma rede real.

Vários padrões de mobilidade foram desenvolvidos para tentar capturar características do movimento e representá-los de uma maneira mais “realística”, ou seja, agregando mais detalhes da realidade modelada. Contudo, ainda não existe um consenso na forma de classificação desses padrões e a literatura apresenta várias propostas diferentes. Em [Cam02] os padrões de mobilidade são classificados em dois tipos: individuais e de grupo. Os padrões de mobilidade individuais representam o comportamento independente do movimento do nodo móvel, enquanto os padrões de mobilidade de grupo são usados quando um grupo de nodos possui um comportamento comum na escolha de seus movimentos.

Uma outra representação é apresentada em [Bet01]. Essa representação possui os seguintes componentes: (i) - nível de aleatoriedade (determinística, híbrida ou aleatória); (ii) - nível de detalhamento (micro mobilidade, macro mobilidade (movimentação individual ou em grupo)); (iii) - representação por simulação ou por modelagem analítica; (iv) - quantidade de dimensões da representação (1D, 2D ou 3D); e (v) - regras de borda quando uma abordagem aleatória é utilizada. A representação apresentada pode ser aplicada tanto em redes sem infra-estrutura, como DTN, quanto em redes infra-estruturadas (redes locais sem fio e de telefonia celular).

Outra classificação para os padrões de mobilidade é apresentada em [Sar07]. Essa classificação propõe quatro modelos: randômicos (o movimento dos nodos é randômico e sem restrições), com dependência temporal (o movimento dos nodos é afetado por seus movimentos passados), com dependência espacial (movimentos correlacionados, equivalente a mobilidade de grupo proposta em [Cam02]) e com dependência geográfica (o movimento dos nodos é limitado por ruas, auto-estradas, obstáculos, entre outros).

Mais recentemente, [Leg06] classifica os padrões de mobilidade em macroscópicos e microscópicos. Os padrões macroscópicos consideram o deslocamento das entidades móveis (pedestres, veículos, animais) no contexto de grandes áreas geográficas. Por outro lado, os padrões microscópicos são projetados para descrever o movimento exato dos nodos em pequenas regiões.

Além disso, autores como [Mus07] e [Yoo06] classificam os padrões de mobilidade em dois tipos: sintéticos e realísticos. Os padrões considerados sintéticos são aqueles onde a simulação ocorre em uma área aberta, sem obstáculos, e cada nodo pode escolher parâmetros de movimentação de forma randômica. Segundo [Ven06], os padrões de mobilidade sintéticos são bastante utilizados nas pesquisas de redes sem fio em geral, como em [Bet03, Bet04, Bro98, Haa01, Lin04, Mus05, Nil04, Spy08]. Contudo, apesar de serem bastante utilizados, esses padrões geralmente não refletem situações do mundo real a respeito da mobilidade dos nodos móveis [Ven06]. Na prática, os nodos

móveis, localizados em qualquer posição geográfica, não se movem de maneira randômica como descrito pelos padrões sintéticos, mas selecionam um destino específico e seguem um caminho para chegar a esse destino.

Como exemplos de padrões sintéticos, pode-se citar o *Random Walk* [Bet01] [Cam02], a versão probabilística do *Random Walk* [Cam02] [Chi98], o *Random Waypoint* [Bet03] [Cam02], o *Random Direction* [Gue87] [Nai05], o *Gauss-Markov* [Cam02] e o *Boundless Simulation Area* [Cam02].

No *Random Walk* o nodo móvel se move de uma localização para outra através de um processo randômico de escolha de direção, medida em graus e escolha de velocidade [Bet01] [Cam02]. O nodo se move na direção e na velocidade selecionada por um período de tempo específico ou por uma distância e então repete o processo. Se o nodo chega no limite da região representada, ele troca o ângulo do movimento e continua pelo novo caminho.

A versão probabilística do *Random Walk* usa uma cadeia de Markov para modelar o movimento do nodo, a qual inclui memória no comportamento do movimento [Cam02]. Nesse modelo duas cadeias de Markov são usadas, uma para modelar o movimento no eixo x e outra para modelar o movimento no eixo y . Este modelo é considerado adequado para representar pessoas e carros em movimento.

No *Random Waypoint*, os nodos móveis escolhem um destino e movem-se até esse destino com uma velocidade constante [Bet03]. Quando o destino é alcançado, o nodo móvel faz uma pausa por um tempo, então escolhe aleatoriamente um novo destino e começa todo o processo novamente com as mesmas regras.

No padrão de mobilidade *Random Direction* a escolha do destino é feita através de uma nova direção, uma velocidade e um tempo de duração para a viagem [Gue87] [Nai05]. Um aspecto importante desse padrão de mobilidade é o comportamento do nodo quando o mesmo alcança a borda, o que é chamado de comportamento de borda. Em [Bet01], são apresentadas três regras de borda diferentes: *Bounce*, *Delete and Replace* and *Wrap Around*. Essas regras de borda garantem que o número de nodos se mantém constante durante toda a análise realizada.

No padrão de mobilidade *Gauss-Markov* a mudança dos movimentos ocorre via um parâmetro no tempo [Cam02]. Inicialmente cada nodo assume uma velocidade e uma direção. Em intervalos fixos de tempo, n , o nodo se movimenta e ocorre a atualização de sua velocidade e direção. A cada intervalo de tempo a próxima localização é calculada baseada na sua localização atual, velocidade e direção do movimento.

O padrão de mobilidade *Boundless Simulation Area* se diferencia dos outros padrões citados anteriormente em função da forma de manipulação da borda da área simulada. Nesse padrão, os nodos móveis que atingirem a borda da área continuam sua trajetória e reaparecem no lado oposto da mesma [Cam02]. Sendo assim, é criada uma área permitindo que os nodos a percorram sem quaisquer obstruções. Além disso, existe um relacionamento entre a direção prévia e velocidade do nodo móvel com sua direção e velocidade atual. A cada t períodos de tempo ocorre a atualização da velocidade e direção.

Já os padrões considerados realísticos consideram mais detalhes da realidade para obter os parâmetros necessários para o movimento dos nodos. Um exemplo é a inserção de obstáculos no modelo, para simular uma cidade com prédios, ruas, casas e vegetação. Na maioria dos casos, a seleção do caminho é influenciada por vias, calçadas e obstáculos, como por exemplo, em um Campus de uma universidade, onde pedestres tendem a utilizar caminhos que interconectam os prédios. Enquanto certos pedestres podem desviar-se desses caminhos (por gramados, por exemplo), a maioria dos indivíduos se move ao longo dos caminhos existentes. Além disso, normalmente, os destinos não são randômicos, mas prédios, restaurantes, estacionamentos e outras localizações específicas, selecionadas conforme a atividade a ser executada pelos pedestres [Sri05]. Portanto, uma série de aspectos deve ser obtida da realidade para fornecer padrões de mobilidade mais realísticos. Contudo, a obtenção desses aspectos pode ser feita de várias formas e variam conforme o trabalho realizado. De acordo com [Paz07], os padrões de mobilidade publicados recentemente são um passo significativo na direção de modelos mais representativos, pois eles consideram um conjunto maior de características do mundo real em suas abordagens, se comparados aos modelos sintéticos.

Como exemplos de padrões realísticos, pode-se citar o *Shortest Path Map Based Movement* [Ekm08] e o *Map Based Movement* [Ekm08], sendo que em ambos os nodos se movimentam levando em consideração informações que são importadas de um mapa de uma cidade (assim como ocorre com um padrão de mobilidade definido em [Ste05]). Essas informações podem restringir o movimento dos nodos em ruas e estradas. Além disso, diferentes grupos de nodos podem ser configurados para usar apenas algumas partes do mapa, permitindo assim a distinção entre automóveis e pedestres por exemplo, e fazendo com que o movimento dos automóveis não ocorra em calçadas ou dentro de prédios. O padrão de mobilidade *Shortest Path Map Based Movement* é derivado do *Random Waypoint* e usa o algoritmo de Dijkstra para encontrar o caminho mais curto que precisa ser viajado a partir da localização corrente do nodo até um destino escolhido randomicamente. Uma vez o nodo alcança seu destino, ele faz uma pausa e um novo destino é também escolhido randomicamente. Nesse modelo, o mapa também pode conter pontos de interesse, ao invés de selecionar posições randômicas para o próximo destino. Pontos de interesse podem ser usados para modelar lojas, restaurantes e atrações turísticas. Já o outro padrão de mobilidade, *Map Based Movement*, é baseado no *Random Walk* e utiliza caminhos pré-definidos dentro do mapa, sendo que diferentes tipos de caminhos podem ser definidos. O destino também é escolhido de forma randômica, mas ao invés de utilizar o caminho mais curto para alcançá-lo, um caminho pré-definido é utilizado.

A utilização de caminhos pré-definidos também é apresentada em [Jar03] e [Mae05], sendo que ambos incluem obstáculos no modelo para tornar mais realístico o movimento dos nodos. Esses obstáculos podem modelar cenários como prédios, cidades, rodovias, etc. Segundo [Jar03], quando o nodo transmite, os obstáculos podem bloquear a propagação da transmissão naquela área, sendo que a transmissão pode ser bloqueada completamente pelo obstáculo. Uma extensão ao padrão apresentado em [Mae05] é definido em [Jar05].

Outro estudo que modela obstáculos é apresentado em [Ven06]. Neste estudo, os obstáculos podem representar prédios, árvores, etc, e bloqueiam o movimento dos nodos, bem como a transmis-

são do sinal. Além disso, o nodo móvel seleciona um destino específico e se move até esse destino. Seguindo uma abordagem baseada em redes sociais, o processo de seleção apresentado em [Ven06] é devido à interação social e demanda da comunidade, e é diferente em diferentes localizações e tempos. Além disso, há um tempo de pausa quando o nodo chega no destino, porém esse tempo não é randômico, ele segue uma distribuição específica do usuário em diferentes localizações. A velocidade do nodo é associada com o grupo no qual o nodo pertence, e ela pode ser alterada se o nodo trocar de grupo. Se um nodo encontra um obstáculo no seu caminho, ele passa ao redor do obstáculo e continua seu movimento até o destino.

Em [Yoo06] é apresentado um Modelo de Mobilidade Estatístico (SMM - *Statistical Mobility Model*), que gera um padrão de mobilidade sintético com dados extraídos de *traces* de pontos de acesso, ou seja, os dados dependem de uma rede cabeada pré-existente. O sistema produz duas saídas: um conjunto de probabilidades de transição e densidades do usuário. A densidade do usuário é usada pelo modelo para gerar as posições iniciais dos usuários móveis, bem como seus destinos. Quando o usuário chega no destino, ele faz uma pausa e seleciona um outro destino a partir da densidade do usuário, e o processo se repete. As probabilidades de transição são usadas para gerar o movimento individual de um usuário. Essa probabilidade é usada em conjunto com algumas rotas, que indicam o caminho mais curto, calculadas pelo modelo e que foram mais acessadas. A velocidade dos nodos é adicionada como um componente externo, independente dos *traces* de dados, gerando uma velocidade não realística. Os *traces* incluem informações sobre os *logs* do sistema, tal como placas de redes sem fio que se associaram, autenticaram, re-associaram, desassociaram, etc. Os *traces* filtrados produzem trajetórias com origem, destino e possíveis localizações intermediárias.

2.4 Roteamento em Redes Tolerantes a Atrasos

Os protocolos de roteamento projetados para redes DTN podem ser definidos de acordo com o grau da informação disponível sobre a topologia da rede. Nesse sentido, segundo [Zha06], as redes DTN podem ser divididas em determinísticas e estocásticas. Nas redes determinísticas, o movimento dos nodos é conhecido a priori, ou seja, é possível prever quando os contatos entre os nodos ocorrem e conseqüentemente sobre as mudanças da topologia em qualquer instante de tempo. Dessa forma, os protocolos de roteamento projetados para redes DTN determinísticas podem escalonar o repasse das mensagens, de forma que as jornadas possam ser determinadas antecipadamente [BX03]. Já nas redes DTN estocásticas, o movimento dos nodos não é conhecido ou previsto. Neste caso, como não existe conhecimento prévio sobre as mudanças topológicas da rede, diferentes estratégias podem ser utilizadas no repasse das mensagens, sendo que esquemas de única e múltiplas cópias podem ser empregados. A distribuição das mensagens para um grande número de nodos irá aumentar a probabilidade de entrega da mesma ao destino, apesar de mais recursos do sistema serem consumidos [Abd07].

Diversos trabalhos que propõem formas de roteamento para redes DTN estocásticas são apresentados na literatura, entretanto o problema do roteamento para essas redes ainda não está resolvido.

A estratégia de repasse mais simples é a da transmissão direta [Abd07], sendo que não ocorre repasses para nodos intermediários, neste caso. Uma outra estratégia de repasse para redes DTN estocásticas que pode ser empregada é a baseada em *flooding*. Nesse caso, haverá a distribuição de cópias das mensagens para todos os nodos encontrados, semelhante ao que ocorre em [Vah00]. Segundo [Spy08], embora o *flooding* possa ser bastante rápido em alguns cenários, o *overhead* envolvido em termos de largura de banda, espaço em *buffer* e consumo de energia é frequentemente proibitivo para dispositivos pequenos, como sensores, por exemplo.

Como limitação do *flooding*, existem técnicas que fazem um controle do número de cópias das mensagens na rede. Esse controle ocorre no momento do repasse, sendo que somente uma quantidade determinada de cópias das mensagens pode ser repassada, como ocorre em [Spy08] e [Tou09], por exemplo. Existem também estratégias, como [Bur05], [Bur06], [Jon05], [Lin03b], [Mus05], [Oli09c] e [Tan03] que decidem o repasse com base em informações históricas sobre o comportamento dos nodos, tentando aumentar a probabilidade de entrega. Além disso, características da rede ou dos nodos podem ser consideradas no repasse, como em [Abd07] e [Bal07], sendo que estratégias híbridas também são utilizadas [Whi09].

Algumas dessas estratégias podem utilizar informações topológicas da rede [Zha06]. Normalmente, nessas estratégias o custo de manutenção das informações topológicas em termos de comunicação e armazenamento é bastante grande, uma vez que os mesmos acumulam informações sobre os nodos da rede e rotas para alcançá-los.

A seguir estão descritas diversas estratégias de roteamento encontradas na literatura.

2.4.1 Transmissão Direta

A transmissão direta é o esquema de roteamento mais simples e de única cópia: depois de gerar uma mensagem, a origem espera até que ela entre em contato com o destino [Abd07]. Como a entrega da mensagem depende de um contato direto entre origem e destino, o atraso na entrega se torna ilimitado. Contudo, sua principal vantagem é a quantidade mínima de transferências para que a mensagem seja entregue.

2.4.2 Epidêmico

Em termos de *overhead* de mensagens, a abordagem oposta à transferência direta é o *flooding*. Nessa abordagem, todos os nodos que recebem uma mensagem repassam-a para todos os seus vizinhos. Contudo, em redes conectadas intermitentemente, simplesmente fazer o repasse das mensagens em *broadcast* não garante que as mesmas cheguem ao destino.

Em [Vah00] foi definido um tipo de roteamento chamado “Epidêmico”, com o objetivo de rotear mensagens em redes DTN maximizando a taxa de entrega, minimizando a latência e a quantidade de recursos consumidos na entrega das mensagens. O funcionamento desse protocolo é semelhante a uma doença epidêmica, enquanto houver espaço em *buffer* mensagens vão sendo espalhadas pela rede como se fosse uma epidemia.

Nesse tipo de protocolo, um *flooding* é feito na rede na tentativa de enviar a mensagem ao destino. Cada nodo possui um *buffer* consistindo de mensagens que o mesmo originou bem como mensagens recebidas de outros nodos, sendo que o tamanho do *buffer* é considerado infinito. Para aumentar sua eficiência, uma tabela *hash* indexa essa lista de mensagens, onde cada mensagem possui um identificador único. Cada nodo armazena um vetor de bits, chamado de *summary vector*, que é uma versão compacta de todas as mensagens armazenadas por um nodo.

Quando dois nodos estão no mesmo alcance de transmissão, aquele com menor identificador inicia uma sessão com o outro, com identificador maior. Para evitar conexões redundantes, cada nodo mantém uma lista dos nodos com quem se conectou recentemente. Assim, a sessão não é reinicializada com aqueles nodos remotos que foram contatados dentro de um determinado período de tempo.

Durante a sessão, os dois nodos trocam seus *summary vectors* para determinar quais mensagens armazenadas remotamente ainda não foram enviadas pelo nodo local, então cada nodo requisita cópias das mensagens que ele ainda não possui. Esse nodo possui autonomia total em decidir quando ele aceitará uma mensagem, como por exemplo, ele pode determinar que não carregará mensagens maiores do que um determinado tamanho ou destinadas para certos nodos [Vah00].

De acordo com [Vah00], o projeto do protocolo Epidêmico associa um único identificador, um contador de *hop* e um ACK opcional para cada mensagem. O identificador é um número único inteiro de 32 bits. Esse identificador é uma concatenação da identificação do nodo e uma identificação da mensagem gerada localmente (de 16 bits). O contador de *hop* determina o número máximo de trocas epidêmicas que uma mensagem pode fazer. Valores grandes de contador de *hop* distribuirão uma mensagem por uma extensão maior da rede, o que pode reduzir o tempo médio de entrega e aumentar o consumo de recursos totais na entrega da mensagem.

De acordo com [Spy08], o roteamento de muitas cópias em paralelo pode ter efeitos desastrosos no desempenho da rede. Em relação ao grande número de transmissões, esquemas baseados em *flooding* começam a sofrer contenções conforme o tráfego aumenta, e seu atraso cresce rapidamente.

Embora o roteamento epidêmico possa ter uma alta taxa de entrega e baixo atraso, ele requer que os nodos tenham um espaço em *buffer* suficientemente grande e pode ocorrer um *overhead* muito grande de mensagens transferidas. Portanto, existem abordagens alternativas para controlar esse *flooding*.

2.4.3 Spray and Wait

Em [Spy08], foi proposta uma família de protocolos de múltiplas cópias chamada de *Spray*. Seu objetivo é gerar um pequeno número de cópias a fim de assegurar que o número de transmissões seja pequeno e controlado. Então, um algoritmo de única cópia é usado para rotear as cópias independentemente.

Um dos protocolos propostos em [Spy08] é chamado de *Spray and Wait* e consiste de duas fases:

- fase *spray*: para cada mensagem que se origina em um nodo origem, C cópias são repassadas para C relays distintos;
- fase *wait*: se o destino não for encontrado na fase *spray*, cada um dos C nodos que carregam uma cópia da mensagem fazem uma transmissão direta para o destino.

A ideia do protocolo *Spray and Wait* é repassar um número suficiente de cópias das mensagens de forma que garanta que, no mínimo, uma delas alcance o destino com alta probabilidade.

Segundo [Spy08], o roteamento *Spray and Wait* não tem um bom desempenho para cenários onde a mobilidade é pequena e localizada. Portanto, foi proposto um esquema onde um número fixo de cópias é espalhado inicialmente, mas cada cópia é roteada independentemente de acordo com um esquema baseado no tempo do último encontro com o destino e com transitividade. Cada nodo armazena o tempo que ocorreu o contato com os outros nodos da rede, sendo que esse tempo é difundido de forma transitiva na rede. Esse esquema é chamado de *Spray and Focus* e consiste de duas partes:

- fase *spray*: para cada mensagem que se origina em um nodo origem, C cópias são repassadas para C relays distintos;
- fase *focus*: cada *relay* pode repassar sua cópia usando um esquema de única cópia baseado no tempo do último encontro com o destino.

Uma definição importante a ser feita tanto para o *Spray and Wait* quanto para o *Spray and Focus* é a quantidade de cópias que precisa ser repassada. Um esquema adotado por [Spy08] é o *spraying* binário. Nesse esquema, o nodo origem da mensagem inicialmente começa com C cópias. Qualquer nodo que possui $C > 1$ cópias da mensagem e encontra outro nodo sem cópias dessa mensagem, repassa $\lfloor C/2 \rfloor$ de suas cópias e mantém $\lceil C/2 \rceil$, até que haja somente uma cópia. Então ele troca para o roteamento de transmissão direta ou para o esquema de repasse que considera o tempo de encontro, dependendo do tipo de roteamento *spray* usado (*wait* ou *focus*).

De acordo com [Spy08], durante a fase de *spray* as mensagens são repassadas de acordo com as seguintes regras:

- cada nodo mantém um *summary vector* com a identificação de todas as mensagens que ele armazena; quando dois nodos se encontram, eles trocam seus vetores e verificam quais mensagens têm em comum (como o roteamento Epidêmico);
- se um nodo (origem ou *relay*) carrega uma cópia da mensagem e tem $C > 1$ cópias da mensagem para repassar, ele a repassa $\lfloor C/2 \rfloor$ de suas cópias para o segundo nodo;
- quando um nodo tem apenas uma cópia da mensagem para repassar, então ele pode somente repassá-la com a regra *wait* ou *focus*.

Segundo [Spy08], foram realizados experimentos com os seguintes protocolos: Epidêmico, *Randomized flooding*, *Utility-based flooding*, *Spray and Wait* binário, *Spray and Focus* e *Spray and Focus* utilizando única cópia. Os resultados mostraram que o protocolo *Spray and Wait* apresentou, com baixo tráfego (200 mensagens em até 10000 unidades de tempo de simulação), uma probabilidade de entrega superior aos outros protocolos, com um atraso semelhante ao protocolo Epidêmico e com um menor número de cópias das mensagens na rede. Já com alto tráfego (1000 mensagens em até 2000 unidades de tempo de simulação), o protocolo se mostrou com níveis de desempenho superiores aos demais protocolos analisados.

2.4.4 Density-Aware Spray-and-Wait

O *Density-Aware Spray-and-Wait* (DA-SW) [Tou09] é uma variação do *Spray and Wait* e tenta de forma dinâmica e baseada em informações locais definir o número de cópias das mensagens que podem ser repassadas na rede para alcançar um atraso médio constante.

O DA-SW é uma versão distribuída do *Oracle-based Spray-and-Wait* (O-SW). A ideia por trás da estratégia é explorar a variação da densidade da rede. Um número maior de cópias de uma mensagem é repassado quando a topologia da rede é esparsa e um número menor de cópias é repassado quando a topologia se torna mais densa. O controle da densidade é feito de acordo com o nível de conectividade atual do nodo.

Uma comparação do O-SW com o *Spray and Wait* foi apresentada em [Tou09]. Os resultados mostraram que o O-SW conseguiu controlar o atraso médio e reduzir o *overhead* [Tou09]. Além disso, foi apresentada uma comparação do O-SW com o DA-SW. Essa comparação mostrou que o atraso gerado pelo DA-SW não foi tão constante quanto o gerado pelo O-SW, devido ao tempo que o DA-SW leva para obter as informações sobre as mudanças topológicas da rede. Contudo, mesmo havendo uma variação no atraso, o *overhead* gerado por ambas estratégias se mostrou bastante próximo.

2.4.5 Shortest Expected Path Routing - SEPR

O *Shortest Expected Path Routing* (SEPR) foi proposto por [Tan03] e estima a probabilidade do repasse com base em dados históricos. Cada nodo mantém um mapa da topologia da rede e os nodos recalculam a probabilidade de encontrar outros nodos quando eles estão conectados. A probabilidade é calculada usando o tempo que dois nodos estão conectados dividido pela janela de tempo de amostras [Tan03].

Cada nodo mantém uma tabela de probabilidade de contato com os demais nodos conhecidos da rede. A cada encontro, essa tabela é trocada e as probabilidades de contato são alteradas. As entradas dessa tabela possuem números de sequência para indicar a “idade” das mesmas. Através desses processos de troca, as entradas da tabela com números de sequências maiores (mais novas) atualizam aquelas com números menores (mais antigas). Dessa forma, as atualizações das probabilidades de contatos são propagadas para todos os nodos.

Durante a recepção de mensagens de atualização da probabilidade do contato, um nodo atualiza sua tabela local e usa o algoritmo de Dijkstra para calcular o tamanho do caminho.

Para cada mensagem armazenada em um nodo é assumido um tamanho de caminho efetivo (EPL - *Effective Path Length*), que é configurado como infinito quando a mensagem é armazenada pela primeira vez. Sempre que uma mensagem é propagada para outro nodo, o EPL é atualizado se o tamanho do caminho desse nodo até o destino for menor do que o seu valor; sendo menor, a mensagem é propagada e um novo valor de EPL é assumido.

O EPL rastreia a história de entrega de uma mensagem [Tan03]. Quanto menor seu valor, maior será a probabilidade de entrega da mensagem ao destino. Além disso, seu valor também é usado para determinar para quais nodos a mensagem deve ser repassada. SEPR utiliza um tamanho de *buffer* fixo e em caso de *overflow* as mensagens com menor EPL são descartadas primeiro.

Segundo [Tan03], o SEPR foi comparado com o protocolo Epidêmico e suas variações, e resultados de experimentos apresentaram uma melhora na taxa de entrega, bem como uma redução no número de mensagens na rede.

2.4.6 Probabilistic Routing Protocol using History of Encounters and Transitivity - P_{Ro}PHET

Uma outra abordagem de repasse de mensagens é chamada de *Probabilistic Routing Protocol using History of Encounters and Transitivity* (P_{Ro}PHET) [Lin09]. Essa abordagem usa uma métrica probabilística que indica se um nodo será capaz de entregar uma mensagem para o outro nodo, sendo chamada de “previsibilidade de entrega”.

Assim como no roteamento epidêmico, os nodos possuem um *summary vector* que indica, de forma compacta, todas as mensagens armazenadas por eles. Quando dois nodos se encontram, eles trocam o *summary vector* e também um vetor de previsibilidade de entrega, contendo a informação de previsibilidade de entrega para os destinos conhecidos pelos nodos. Essa informação é usada para decidir se o repasse de uma mensagem para um certo nodo deve ser feito ou não. Caso nenhum valor esteja armazenado para um dado destino, a previsibilidade de entrega para esse destino é considerada zero [Lin09].

O cálculo da previsibilidade de entrega possui três passos [Lin09]: quando dois nodos se encontram, o primeiro passo é atualizar a previsibilidade de entrega para o outro nodo, então os nodos que se encontram com frequência possuem uma alta previsibilidade de entrega. Como segundo passo, se um par de nodos não se encontra durante um intervalo de tempo, ele possui uma probabilidade baixa de repassar mensagens entre eles. Nesse caso, a previsibilidade de entrega deve envelhecer, sendo reduzida. O terceiro passo do cálculo da previsibilidade de entrega diz respeito à propriedade de transitividade, que é baseada na observação de que se um nodo A encontra com frequência um nodo B, e esse nodo B encontra com frequência um nodo C, então provavelmente o nodo C é um bom nodo para repassar mensagens destinadas ao nodo A.

Segundo [Lin09], quando um vizinho é detectado, uma mensagem de HELLO é enviada para o mesmo. Após, é enviado o *summary vector* e é atualizada a previsibilidade de entrega. Feito isso, o nodo verifica quais mensagens devem ser enviadas para o vizinho detectado e quais ele irá armazenar

para repasse futuro. O nodo envia as mensagens para o vizinho se esse possuir maior previsibilidade de entrega [Lin03b]. As mensagens repassadas não são removidas do nodo, mas ficam armazenadas em *buffer* (desde que haja espaço disponível), pois o nodo que as repassou pode encontrar outro melhor ou o próprio destino final no futuro. Caso os *buffers* estejam cheios quando uma nova mensagem for recebida, a mensagem mais antiga deve ser retirada de acordo com o sistema de gerenciamento de fila usado. Nesse trabalho, o sistema usado foi o FIFO (*First In First Out*).

Uma característica dessa estratégia é que os nodos acabam tendo informações sobre os outros nodos da rede a partir da troca do vetor de previsibilidade de entrega.

Em [Lin09] é apresentada uma comparação do P_{Ro}PHET com o protocolo Epidêmico, sendo que os resultados apresentaram uma melhora de taxa de entrega, com uma menor quantidade de mensagens sendo repassadas e um maior atraso em alguns cenários simulados.

2.4.7 Meeting Visit - MV

O objetivo do protocolo de roteamento MV (*Meeting Visit*) [Bur05] é aprender a frequência de encontro entre pares de nodos e suas visitas em certas regiões geográficas. As frequências passadas são usadas para classificar cada mensagem de acordo com sua probabilidade de entrega através de um caminho específico. É assumido que a probabilidade de visitar uma região no futuro está fortemente relacionada com a história de visitas do nodo naquela região.

O algoritmo de repasse do protocolo MV funciona da seguinte forma: quando dois nodos se encontram, trocam entre si uma lista das mensagens que eles carregam, com seus destinos e com a probabilidade do nodo entregar a mensagem ao destino. Os nodos selecionam na lista as mensagens que eles não possuem e retiram aquelas cuja probabilidade de entrega seja mais alto no outro nodo.

Como restrições, em [Bur05] é assumido um tamanho de *buffer* infinito para as mensagens que são originadas pelo próprio nodo e um *buffer* com tamanho limitado para armazenar as mensagens recebidas de outros nodos. Além disso, a largura de banda é considerada como infinita, as mensagens são entregues para destinos localizados em pontos fixos e o número de *hops* é limitado.

Segundo [Bur05], experimentos para avaliar o uso dos *buffers* foram realizados, sendo que a estratégia foi comparada com três algoritmos: sem *buffer*, *buffer* ilimitado e FIFO. Os resultados mostraram que o protocolo MV possui uma taxa de entrega superior aos demais algoritmos, no entanto não foram realizados experimentos com outros protocolos de roteamento para redes DTN.

2.4.8 Context-aware Adaptive Routing - CAR

Semelhante ao P_{Ro}PHET, no protocolo *Context-aware Adaptive Routing* (CAR) [Mus05] o nodo que carrega a mensagem possui uma probabilidade mais alta de entregar mensagens para um destino do que outros nodos. Contudo, uma das diferenças está no fato que ele integra mecanismos síncronos e assíncronos para a entrega das mensagens. Nesse contexto, na entrega síncrona, quando uma mensagem chega, o caminho para o destino existe e a mensagem pode ser repassada usando um protocolo de roteamento pró-ativo existente (o DSDV [Per94], neste caso). Já na entrega

assíncrona, quando chegam as mensagens, um caminho para o destino não pode ser encontrado e as mensagens devem ser armazenadas para um repasse posterior ou serem enviadas para aqueles nodos que possuem a maior probabilidade de entrega ao destino. Esses nodos atuarão como carregadores de mensagens.

Nesse protocolo a informação de contexto é definida como um conjunto de atributos que descrevem os aspectos do sistema que podem ser usados para melhorar o processo de entrega [Mus05]. Cada nodo usa uma predição local para calcular a probabilidade de entrega. O processo de predição é usado durante desconexões temporárias e utiliza informações de contexto a partir da técnica de filtros de Kalman [Mus05].

Apesar da descrição do protocolo informar que as mensagens podem ser repassadas para vários carregadores que possuem as probabilidades mais altas, os experimentos realizados em [Mus05] consideram um cenário onde os nodos podem repassar as mensagens para somente um carregador. Caso o carregador se mova e encontre um outro nodo com probabilidade de entrega mais alta, a mensagem é transferida para esse nodo.

Segundo [Mus05], o protocolo CAR foi comparado, através de experimentos, com o protocolo Epidêmico e com a estratégia de *flooding*. Os resultados dos experimentos mostraram que o protocolo CAR teve uma taxa de entrega entre o *flooding* e o Epidêmico, com um número menor de cópias das mensagens na rede em alguns cenários.

2.4.9 Minimum Estimated Expected Delay - MEED

Em [Jon05] foi proposto um protocolo chamado de *Minimum Estimated Expected Delay* (MEED). Esse protocolo utiliza como métrica o menor atraso fim-a-fim. Tal métrica é calculada utilizando históricos dos contatos observados pelos nodos da rede, que registram o tempo das conexões e desconexões de acordo com uma janela de históricos. O tamanho dessa janela pode ser ajustado em cada nodo.

O protocolo é baseado no algoritmo Dijkstra usado em redes tradicionais, mas com adaptações para ser utilizado em redes DTN. Os nodos tem informações sobre a topologia da rede através de pacotes de estado de enlace que são enviados para todos os nodos da rede pelo protocolo Epidêmico, sendo que rotas são recalculadas sempre que conexões são estabelecidas.

Quando dois nodos se encontram, eles trocam *summary vectors* com informações relativas a todas as tabelas de estado de enlace que receberam [Jon05]. Cada tabela possui um número de sequência, que ajuda a determinar quais são as mais recentes. Após a troca dessas informações, os nodos devem possuir o mesmo estado da topologia, podendo recalculer suas tabelas de roteamento, para então repassar as mensagens armazenadas. Segundo [Jon05], mensagens são repassadas imediatamente quando a métrica configurada nos enlaces for igual a zero, a qual indica que o contato está disponível e o atraso é pequeno.

A desvantagem desta estratégia é o uso de recursos e processamento nos nodos, principalmente em redes grandes, devido à troca de informações sobre a topologia da rede e a manutenção das tabelas de roteamento nos nodos.

MEED considera o uso de *buffers* de tamanho fixo e utiliza a técnica de *drop tail* no descarte das mensagens.

Segundo [Jon05], a avaliação do MEED foi realizada com as seguintes estratégias: Epidêmico, *Earliest Delivery* (ED), *Minimum Expected Delay* (MED) e uma variante do MED. Em um cenário com limitação de *buffer*, o MEED obteve uma taxa de entrega superior ao Epidêmico, contudo ficou abaixo das demais estratégias analisadas. Já em relação ao atraso, no mesmo cenário, seu desempenho foi pior que o Epidêmico, ED e a variante do MED, ficando similar ao MED. Além disso, o MEED não mostrou um bom desempenho quando houve variação na largura de banda, ficando abaixo dos protocolos analisados.

2.4.10 MaxProp

O MaxProp [Bur06] utiliza um mecanismo de repasse baseado em *flooding*, contudo, quando um contato é descoberto, nem todas as mensagens armazenadas pelo nodo serão repassadas.

A decisão de quais mensagens devem ser entregues é feita por uma estimativa da probabilidade de entrega. O Maxprop mantém uma fila ordenada com base no destino de cada mensagem e na probabilidade estimada do caminho até o destino. Para obter essa probabilidade, cada nodo mantém um vetor de tamanho $n - 1$ (onde n é o número de nodos na rede) consistindo da probabilidade do nodo encontrar cada um dos outros nodos na rede. Quando dois nodos se encontram, a probabilidade de entrega para o nodo encontrado é atualizada e os vetores de probabilidade são trocados. Dessa forma, cada nodo terá o vetor atualizado do outro nodo [Bur06].

A partir de todos os vetores recebidos, o nodo calcula o caminho de menor custo até cada um dos destinos das mensagens que o mesmo armazena. O cálculo desse custo leva em consideração as probabilidades de entrega armazenadas nos vetores. O MaxProp envia primeiro as mensagens com o maior número de *hops* percorrido. Além disso, ele considera um *buffer* ilimitado para mensagens originadas pelo próprio nodo e um *buffer* com tamanho fixo para mensagens recebidas de outros nodos. As mensagens com menor probabilidade de entrega são descartadas primeiro em caso de *overflow*.

Uma outra característica do MaxProp é a notificação, por uma mensagem de confirmação, a toda a mensagem entregue ao destino, sendo replicada a todos os nodos da rede. A mensagem de confirmação consiste da criptografia do conteúdo, origem e destino de cada mensagem através do mecanismo de *hash* [Bur06]. Tal mensagem é utilizada para remover dados antigos dos *buffers*.

O MaxProp foi comparado, segundo [Bur06], com outras três estratégias usadas em roteamento: *Oracle-based Dijkstra*, *Drop-Least Encountered* e roteamento randômico. Os resultados mostraram que o MaxProp, em geral, entrega mais mensagens e mantém um menor atraso.

2.4.11 Neighborhood Contact History - NECTAR

O protocolo NECTAR (*Neighborhood Contact History*) proposto em [Oli09d] e melhorado em [Oli09c] baseia-se no histórico de contatos entre vizinhos para decidir sobre o repasse das mensagens,

definindo uma métrica denominada de *Neighborhood Index*. Nesse protocolo, as informações sobre os contatos são trocadas pelos nodos. Tal protocolo fundamenta-se na heurística que nodos em cenários estocásticos movimentam-se de forma que existe uma certa probabilidade que vizinhos possam ser reencontrados [Oli09c].

Com o objetivo de evitar o consumo desnecessário dos recursos da rede, cada nodo mantém, em cache, uma relação de vizinhos que se conectaram recentemente. Durante o período de contato, os nodos efetuam, inicialmente, a transmissão de mensagens cujo destino seja o nodo que estabeleceu o contato. Em seguida, ocorre a troca de informações sobre a vizinhança (*Neighborhood Index*) e, eventualmente, o encaminhamento das demais mensagens armazenadas.

O *Neighborhood Index* é calculado de acordo com os contatos que ocorrem entre os nodos, sendo que seu valor é maior para contatos mais duradouros e frequentes. Quando ocorre o primeiro contato entre dois nodos, seu valor é inicializado com “1” e, à medida que os nodos permanecem em contato, esse valor vai aumentando linearmente [Oli09c]. Além disso, *Neighborhood Index* considera duas métricas distintas em seu cálculo, uma métrica de envelhecimento e outra de distância. A métrica de envelhecimento reduz o valor de *Neighborhood Index* quando os nodos param de se encontrar. Já a de distância representa a quantidade de saltos entre dois nodos. Dessa forma, a fórmula do *Neighborhood Index* favorece o encaminhamento de mensagens para vizinhos que encontrem-se próximos ao destino e que estejam ou tenham estado recentemente em contato com o mesmo [Oli09c]. As mensagens entregues ao destino são eliminadas dos *buffers*, mas uma cópia do cabeçalho é mantida por um tempo, evitando, assim, o recebimento de cópias de mensagens já entregues.

Além disso, segundo [Oli09c], o protocolo NECTAR possui uma fase de operação em que mensagens podem ser replicadas na rede, semelhante ao protocolo Epidêmico. Contudo, para que isso ocorra, uma expressão que leva em consideração certos níveis precisa ser válida e os nodos somente armazenarão mensagens transmitidas de forma epidêmica se a taxa de ocupação da área de armazenamento estiver abaixo de um determinado limiar.

Como política de descarte de mensagens, o NECTAR mantém as mensagens mais recentes e aquelas que foram encaminhadas para poucos nodos. Para tanto, é realizado um controle da quantidade de cópias repassadas, bem como da quantidade de tempo decorrida desde a recepção da mensagem.

Segundo [Oli09c], os protocolos Epidêmico, PRoPHET e NECTAR foram avaliados em diferentes cenários, sendo que o NECTAR obteve uma maior taxa de entrega. Quanto à transmissão de mensagens na rede, o NECTAR obteve um melhor desempenho, exceto quando comparado com o PRoPHET em cenários com maior capacidade de armazenamento. Além disso, as mensagens foram entregues com uma menor quantidade de saltos e em menor tempo que os demais protocolos avaliados.

2.4.12 Core-aided

O protocolo chamado de *Core-aided* [Abd07] foi criado para cenários específicos onde há dois tipos de nodos na rede: os que possuem recursos limitados e aqueles que não possuem limitação de recursos. Essa estratégia usa os nodos que não possuem recursos limitados como *core nodes*. Tais nodos, quando entram em contato com aqueles com recursos limitados, recebem algumas ou todas as mensagens deles. Em [Abd07] foram propostas duas estratégias possíveis *copy-to-core* ou *dump-to-core*. Na primeira estratégia, os nodos apenas fazem uma cópia de suas mensagens para o *core*. Isso aumentará o número de mensagens carregadas na rede, reduzindo o atraso e aumentando a taxa de entrega. Se há mais do que um nodo *core* no sistema, eles podem trocar suas mensagens semelhante ao roteamento Epidêmico, já que os mesmos não são limitados em termos de recursos.

Na segunda estratégia, os nodos removem a mensagem entregue ao *core*. Isso pode ser útil em ambientes onde o tamanho do *buffer* na maioria dos nodos é bastante restrito. Um inconveniente dessa estratégia é um possível aumento do atraso na entrega das mensagens. Isso pode ocorrer porque somente o nodo *core* carrega as mensagens, o que diminui a chance de contato com o nodo destino em um tempo curto.

A avaliação de desempenho do *Core-aided* foi realizada em [Abd07] comparando os resultados teóricos com resultados obtidos através de simulação. Os resultados mostraram que para três métricas analisadas, taxa de entrega, atraso na entrega e ocupação de *buffer*, os resultados foram equivalentes. Contudo, experimentos não foram realizados comparando o desempenho do *Core-aided* com outros protocolos de roteamento para redes DTN.

2.4.13 Resource Allocation Protocol for Intentional DTN - RAPID

O protocolo RAPID (*Resource Allocation Protocol for Intentional DTN*) [Bal07] utiliza informações locais para decidir sobre o repasse das mensagens. Esse protocolo determina, a cada oportunidade de transferência, se a replicação da mensagem justifica o uso dos recursos. O protocolo é baseado na premissa que os algoritmos de roteamento para DTN afetam o desempenho das métricas e tenta intencionalmente otimizar uma determinada métrica, como por exemplo, atraso na entrega ou quantidade de pacotes entregues em um determinado tempo. Para tanto, o RAPID define uma função utilidade (semelhante a uma prioridade), que determina se as mensagens devem ser replicadas. Essa função assume um valor para cada mensagem baseado na métrica sendo otimizada. O RAPID replica primeiro as mensagens que localmente resultam em uma melhora na métrica escolhida.

Assim como o MaxProp, o protocolo RAPID é baseado em *flooding* e tentará replicar todos os pacotes armazenados desde que permitido pela função utilidade. Diferente do MaxProp, o RAPID não propaga mensagens de confirmações através da rede para cada entrega realizada. Ao invés, uma lista de mensagens já entregues é trocada quando ocorre contato entre os nodos. Além disso, cada nodo mantém um *summary vector* com as mensagens que o mesmo armazena. Quando dois nodos se encontram o *summary vector* é trocado entre os nodos e somente mensagens novas são

repassadas. Esse protocolo utiliza um tamanho de *buffer* fixo e retira primeiro mensagens com mais baixa utilidade no caso de *overflow*.

Experimentos foram realizados em [Bal07] comparando o RAPID com os protocolos MaxProp, *Spray and Wait*, PProPHET, *Random* e *Optimal*. Para cada experimento realizado, o RAPID foi configurado para tentar otimizar a métrica sendo avaliada, sendo que seu desempenho se mostrou superior aos demais protocolos. Algumas métricas que foram configuradas em [Bal07] para serem otimizadas no RAPID foram: minimização do atraso médio, minimização do atraso máximo e maximização do número de pacotes entregues.

2.4.14 Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks - HYMAD

O HYMAD (*Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks*) [Whi09] é um protocolo de roteamento para DTN que usa a ideia de grupos de nodos fisicamente próximos. HYMAD combina técnicas de roteamento *ad hoc* tradicional com a abordagem para redes DTN.

De acordo com [Whi09], HYMAD periodicamente procura mudanças na topologia da rede e constrói grupos disjuntos temporários de nodos conectados. HYMAD é completamente descentralizado e somente faz uso das trocas de informações topológicas entre os nodos.

No HYMAD, o roteamento dentro dos grupos é realizado por um algoritmo de vetor de distância simples. Os nodos são dinamicamente agrupados por um algoritmo de particionamento de redes distribuído. Os grupos aceitam novos membros enquanto seu diâmetro for menor do que um parâmetro de diâmetro máximo. Se o diâmetro de um grupo se expande devido a uma falha interna de contato, alguns membros são excluídos para satisfazer a restrição do diâmetro.

No roteamento entre grupos, um nodo de borda cuida da distribuição das mensagens. Esse nodo utiliza um protocolo de *broadcast* periódico para ter informações sobre a composição dos grupos adjacentes, bem como das mensagens que eles armazenam. Quando um nodo de borda aprende que um grupo tem cópias de uma mensagem que um grupo vizinho não possui, ele pode requisitar a custódia de parte das cópias para então repassá-las ao grupo vizinho, semelhante à estratégia utilizada pelo protocolo *Spray and Wait*. Se o destino da mensagem está no grupo vizinho, o nodo de borda requisita a mensagem e repassa-a para o mesmo [Whi09].

Segundo [Whi09], experimentos foram realizados comparando o HYMAD, usando diâmetro máximo igual a “2”, com os protocolos Epidêmico e *Spray and Wait*. Os resultados mostraram que o desempenho do HYMAD ficou entre o Epidêmico e o *Spray and Wait*, quando a probabilidade de entrega das mensagens foi analisada para cinco e vinte cópias.

2.4.15 Comparação entre os Protocolos de Roteamento

A Tabela 2.1 apresenta um resumo comparativo de alguns protocolos de roteamento para DTN apresentados anteriormente. A comparação realizada utilizou as seguintes métricas:

- gerenciamento de *buffers* (escalonamento): indica a técnica de gerenciamento de *buffer* utilizada no envio das mensagens quando dois nodos se encontram. Todos os protocolos analisados priorizam entrega direta para depois enviar as mensagens restantes caso o tempo de contato permita. Especificidades são colocadas na tabela;
- gerenciamento de *buffers* (descarte): indica a técnica de gerenciamento de *buffer* utilizada no descarte das mensagens quando ocorre *buffer overflow*. Quando este aspecto apresentar valor "infinito" na tabela, indica que os experimentos disponíveis na literatura são hipotéticos, supondo *buffers* ilimitados e assim o descarte de mensagens não existe;
- informações que um nodo armazena: indica as informações que os nodos armazenam quando recebem mensagens de outros nodos ou que descobrem quando dois nodos se encontram. Em todos os protocolos os nodos utilizam *buffers* para armazenar as mensagens que precisam ser entregues;
- uso de recursos da rede: indica as informações de controle que são enviadas na rede quando dois nodos se encontram;
- limite do número de cópias: indica se o protocolo controla o número de cópias das mensagens;
- repasse controlado: indica se o protocolo restringe o repasse das mensagens ou se faz um repasse aleatório.

Tabela 2.1: Resumo e comparação entre os protocolos de roteamento.

Protocolo	Gerenciamento de buffers (escolonamento)	Gerenciamento de buffers (descarte)	Informações que um nodo armazena	Uso de recursos da rede	Límite do número de cópias	Repasso controlado	Observações
Epidêmico [Vah00]	Não informado.	Infinito	Lista com nodos que se conectou recentemente.	<i>Summary vector</i>	Não	Não	
Spray and Wait [Spy08]	Não informado.	Infinito	Nenhuma	<i>Summary vector</i>	Sim	Não	
DA-SW [Tou09]	Não informado.	Infinito	Nenhuma	<i>Summary vector</i>	Sim	Não	Número de cópias dinâmico
SEPR [Tan03]	Não informado.	Tamanho de buffer fixo. Retira mensagens com menor EPL (<i>Effective Path Length</i>).	Tabela com probabilidade de contato com todos os nodos da rede.	Tabela com probabilidade de contato com os nodos.	Não	Sim	
PRoPHET [Lin09]	Não informado.	Tamanho de buffer fixo, uso de FIFO.	Vetor de previsibilidade de entrega para todos os nodos da rede.	<i>Summary vector</i> . Vetor de previsibilidade de entrega para os nodos.	Não	Sim	Trabalha com transitividade.
MV [Bur05]	Envia primeiro mensagens com maior probabilidade de entrega.	<i>Buffer</i> infinito para mensagens originadas pelo próprio nodo. <i>Buffer</i> fixo para mensagens recebidas de outros nodos. Retira mensagens com menor probabilidade de entrega.	Lista de mensagens com probabilidade de entrega para cada mensagem.	<i>Summary vector</i> com probabilidades de entrega vinculadas nas mensagens.	Não	Sim	Limita o número de hops das mensagens.
CAR [Mus05]	Não informado.	Tamanho de <i>buffer</i> fixo. Não informa a técnica usado para descartar.	Tabela com probabilidade de entrega para todos os nodos da rede.	Tabela com probabilidade de entrega.	Cópia única.	Sim	Integrado com DSDV. Cálculo da probabilidade de entrega leva em consideração a conectividade dos nodos e o nível de energia.

Continua na próxima página.

Tabela 2.1 Resumo e comparação entre os protocolos de roteamento. Continuação da página anterior.

Protocolo	Gerenciamento de buffers (escalonamento)	Gerenciamento de buffers (descarte)	Informações que um nodo armazena	Uso de recursos da rede	Limite do número de cópias	Repasso controlado	Observações
MEED [Jon05]	Não informado.	Tamanho de <i>buffer</i> fixo, uso de <i>drop tail</i> .	Tabela com o tempo do último contato com todos os destinos.	<i>Summary vector</i> . Informações sobre a topologia da rede.	Não	Sim	Calculam o atraso mínimo esperado. Deriva a topologia da rede.
MaxProp [Bur06]	Envia primeiro mensagens com maior número de <i>hops</i> percorrido.	<i>Buffer</i> infinito para mensagens originadas pelo próprio nodo. <i>Buffer</i> fixo para mensagens recebidas de outros nodos. Retira mensagens com menor probabilidade de entrega.	Tabela com probabilidade de contato com todos os nodos da rede.	Tabela com probabilidade de contato.	Não	Sim	Calculam o custo para um determinado destino. Limita o número de <i>hops</i> das mensagens. Utiliza ACK.
NECTAR [Oli09d]	Envia primeiro as mensagens cujo campo <i>NextHop</i> aponte para o nodo de contato.	Tamanho de <i>buffer</i> fixo. Retira mensagens mais antigas e encaminhadas para mais nodos da rede.	Tabela <i>Neighborhood Index</i> (informação sobre contatos e radouros frequentes).	Tabela com <i>Neighborhood Index</i>	Não	Sim	Mensagens recebem TTL após a entrega (<i>Passive Ack</i>). Utiliza métrica de distância e métrica de envelhecimento.
Core-Aided [Abd07]	Não informado.	Infinito	Não	Não comenta no artigo.	Depende da forma de repasse empregada	Sim (para os cores)	Para cenários com 2 tipos de nodos (com e sem recursos limitados).
RAPID [Bal07]	Envia primeiro mensagens com mais alta "utilidade" (prioridade).	Tamanho de <i>buffer</i> fixo. Retira primeiro mensagens com mais baixa "utilidade" (prioridade).	Depende da métrica sendo otimizada.	<i>Summary vector</i> .	Não	Sim	Utiliza uma função "utilidade", semelhante a um prioridade. É dependente da métrica sendo otimizada. Utiliza ACK.

2.5 Métricas de Avaliação para Redes Tolerantes a Atrasos

Dadas as características das redes DTN, é importante considerar formas significativas para avaliar o comportamento dos protocolos de roteamento para essas redes. Em muitos casos, métricas tradicionais usadas na avaliação de protocolos de roteamento são portadas para ambientes DTN. As métricas mais comuns encontradas na literatura são [Lin09], [Spy08], [Bal07]:

Percentual de Mensagens Entregues: mede a quantidade de mensagens que foram corretamente entregues a todos os destinos;

Overhead: pode se referir a quantidade de mensagens de controle trocadas entre os nodos da rede, bem como ao número médio de cópias de cada mensagem gerado na rede, como em [Mus05] e [Nun09a]. Como as redes DTN possuem poucos recursos, os protocolos de roteamento deveriam ser otimizados para alcançar um baixo *overhead*. Este trabalho considera *overhead* como a segunda definição;

Atraso na Entrega das Mensagens: mede o tempo médio desde a criação de uma mensagem até que a mesma seja entregue ao destino. Normalmente essa medida considera somente as mensagens entregues ao destino.

Em adição a esse conjunto de métricas, é possível fazer uso de métricas mais específicas que ajudam a compreender o comportamento das métricas citadas anteriormente ou a fornecer razões para comportamentos inesperados. Como métricas mais específicas pode-se citar [Nun09b]:

Ocupação de Buffer: mede o número médio de mensagens armazenadas por um nodo, normalmente obtido computando-se a média para todos os nodos;

Mensagens Descartadas devido a Buffer Overflow: mede o número de mensagens descartadas por *buffer* cheio, normalmente considerando uma média de todos os nodos. Essa métrica se correlaciona com a métrica anterior e quantifica as consequências de uma alta ocupação do *buffer*;

Entregas Redundantes: fornece o número médio de cópias extras entregues ao destino, depois que a primeira mensagem já foi entregue. É uma medida de redundância na entrega da mensagem e pode identificar o uso desnecessário de recursos. Ela é computada somente para mensagens entregues;

Tempo de Vida das Mensagens: mostra o tempo médio que uma mensagem permanece ativa na rede, isto é, o tempo desde sua criação até que a última cópia restante seja descartada de um nodo. Ao se analisar esta métrica junto com o atraso na entrega das mensagens, pode-se verificar se as mensagens estão por muito tempo utilizando os recursos dos nodos.

3. Comportamento de Entrega e Atraso em Redes DTN e Ad Hoc

As redes DTN possuem várias características em comum com as redes *ad hoc*, como por exemplo, a mobilidade dos nodos, o roteamento com múltiplos saltos e a independência de infra-estrutura inexistente previamente. A diferenciação entre ambas acontece principalmente por características como inexistência de um caminho fim-a-fim entre origem e destino, longos e variados atrasos e altas taxas de erro que ocorrem nas redes DTN. Em especial, os nodos das redes DTN estocásticas, foco deste trabalho, não possuem informações sobre as características de mobilidade dos outros nodos da rede, sendo ainda mais difícil definir como o repasse das mensagens deve ser feito.

As disrupções que ocorrem estão diretamente relacionadas com a mobilidade e a baixa densidade da rede. A densidade da rede é influenciada pela quantidade de nodos na mesma, bem como pelo raio de alcance utilizado nas transmissões. Em redes mais densas, mesmo havendo a mobilidade dos nodos, é mais provável que haja um caminho até o destino do que em redes mais esparsas. Contudo, não foram encontrados estudos quantitativos que consideram a densidade da rede na comparação entre redes DTN e *ad hoc*. Somente há trabalhos que analisam em separado os dois tipos de redes, como em [Bro98], [Che07], [Sha08], [Tsu03] e [Zha07].

Este Capítulo apresenta um estudo que ajuda a compreender como a variação da densidade de uma rede influencia no comportamento de entrega e atraso em redes DTN e *ad hoc*, tentando identificar as potencialidades das diferentes tecnologias. Esse estudo é importante, pois ajuda a caracterizar o tipo de rede em estudo para a utilização de diferentes estratégias de roteamento.

Nesse sentido, um estudo experimental foi conduzido comparando o desempenho de um protocolo de roteamento para redes DTN com um protocolo usado em redes *ad hoc*. O protocolo de roteamento para redes DTN utilizado nos experimentos é proposto neste trabalho e é chamado de APRP-Group - vide Capítulo 6. Tal protocolo foi utilizado devido às características de desempenho que o mesmo apresenta em comparação com outros protocolos presentes na literatura. O APRP-Group foi projetado para trabalhar com tipos de nodos diferentes (com diferentes parâmetros de mobilidade), mas também apresenta um bom desempenho quando a rede possui apenas um tipo de nodo. Já o protocolo de roteamento para redes *ad hoc* utilizado foi o AODV (*Ad hoc On-demand Distance Vector*) [Per03]. Esse protocolo foi escolhido por ser um protocolo padronizado pelo IETF (*Internet Engineering Task Force*) [Per03], além de ser um dos mais difundidos e estudados na literatura [Bro98, Chl03, Liu05]. O AODV é considerado um protocolo reativo, ou seja, o estabelecimento de rotas ocorrerá sob demanda.

A seguir encontram-se descritos os parâmetros utilizados no experimentos, bem com os resultados encontrados.

3.1 Parametrização dos Experimentos

Os experimentos foram realizados com dois simuladores, o ns-2 (*Network Simulator*) [Fal09], bastante utilizado em análises de redes *ad hoc* [Bro98], [Sha08], [Tsu03], e o ONE (*Opportunistic Network Environment*) [Ker07], específico para DTN e em crescente utilização pela comunidade acadêmica [Fer09], [Hua09], [Lee09], [Pol09], [San08].

Os dois simuladores foram configurados com os mesmos parâmetros para gerar comparações coerentes. O movimento dos nodos, bem como a carga de trabalho usada no simulador ONE foram convertidos para serem utilizados no simulador ns-2. A conversão do movimento dos nodos foi realizada por uma ferramenta própria do ONE, que gera um arquivo no padrão usado pelo ns-2. A carga de trabalho usada no ONE é gerada a partir de outra ferramenta que vem com o próprio simulador. Essa recebe como parâmetro o intervalo de tempo em que as mensagens devem ser criadas, o número de mensagens, o tamanho e o intervalo dos nodos que serão origem e destino das mensagens. Como saída, a ferramenta gera uma lista com informações sobre cada mensagem, como por exemplo, o instante de tempo de sua criação, o nodo que deve criar a mensagem, o nodo que deve recebê-la e o tamanho da mesma. Essa lista foi convertida para o ns-2 com uma ferramenta própria, assumindo que cada mensagem DTN equivale a uma mensagem UDP (*User Datagram Protocol*) de mesmo tamanho, com mesma origem, destino e tempo de criação, tendo CBR (*Constant Bit Rate*) como aplicação.

Um número de 1000 mensagens de 64 bytes foi criado em um intervalo entre 1000 e 10000 segundos em tempo de simulação, sendo que qualquer nodo presente na simulação poderia aleatoriamente ser origem ou destino das mensagens.

A área simulada foi configurada com um tamanho de 4500 m x 3400 m. Este é o tamanho da área central da cidade de Helsinki, na Finlândia, sendo que um mapa dessa área, que vem com o simulador ONE, será utilizado em experimentos realizados nos Capítulos 4, 5 e 6. O número de nodos variou entre 30, 50, 100 e 300 nodos, sendo usados como forma de variar a densidade da rede.

Os nodos se movimentaram de acordo com o padrão de mobilidade *Random Waypoint* (descrito na Seção 2.3), com velocidade de 1,0 m/s e com tempo de pausa de 60 segundos, indicando pouca mobilidade dos nodos na rede. A transmissão das mensagens ocorreu com uma velocidade de 2Mbps e os nodos foram configurados com um tamanho de *buffer* de 64Kbytes no ONE e 1000 pacotes no ns-2. O TTL (*Time to Live*) das mensagens foi considerado infinito. Além disso, o raio de alcance utilizado variou em 20 m, 40 m, 60 m, 80 m e 250 m, sendo esta outra forma de variar a densidade da rede.

O ns-2 utiliza um raio padrão, de acordo com o tipo de interface de rede sem fio configurada. O tipo de interface configurada foi o "WirelessPhy"¹, que tem um raio de 250 m como padrão. Para alterar o valor do raio, foi necessário utilizar uma ferramenta para essa função. Tal ferramenta vem com o ns-2 e, ao executá-la, é necessário passar como parâmetro o raio de alcance que se

¹Atua como uma interface de hardware através da qual o nodo acessa o canal [Fal09].

deseja utilizar, bem como o modelo de propagação utilizado. Como saída é retornado um valor que indica a sensibilidade da antena, dado em dB, o qual deve ser configurado no *script* do ns-2. O modelo de propagação utilizado nos experimentos foi o “TwoRayGround”². Além disso, como tipo de canal, utilizou-se o “WirelessChannel”³, como tipo de antena foi configurado o “OmniAntenna”⁴ e como tipo de MAC (*Medium Access Control*) utilizou-se o “802_11”⁵. Esses parâmetros não são necessários na configuração do ONE.

Além dessas configurações, o protocolo APRP-Group também precisa ser configurado. Os parâmetros e valores necessários na sua configuração estão descritos no Capítulo 6.

Os experimentos para a rede *ad hoc* rodaram até 1100 segundos em tempo de simulação, 100 segundos a mais depois da última mensagem ser criada. Enquanto para a rede DTN, os experimentos rodaram até 110000 segundos em tempo de simulação. Esses tempos foram configurados com valores diferentes porque pretende-se analisar o tempo de entrega da quase totalidade das mensagens. Nas redes *ad hoc*, no momento em que uma mensagem é criada, o nodo tenta estabelecer uma rota até o destino. Caso a rota não possa ser estabelecida, a mensagem é descartada do *buffer* do nodo que a criou. Portanto, a adição de 100 segundos depois da última mensagem criada é suficiente para que a mesma seja entregue caso haja uma rota, de acordo com observações realizadas. Já nas redes DTN, o mecanismo *store-carry-forward* é utilizado, fazendo com que os nodos armazenem a mensagem em *buffer* até que o repasse da mesma ocorra. Dessa forma, um tempo maior de simulação foi configurado de maneira que a maioria das mensagens pudesse ser recebida pelos nodos, como ocorre em uma rede real.

Para cada cenário simulado, foram realizadas sete rodadas de simulação. Os resultados apresentados nesta seção foram obtidos pela média de todas as rodadas. Para cada valor apresentado, as barras verticais associadas representam um intervalo de confiança de 95%. Contudo, algumas vezes o intervalo de confiança está bastante próximo do valor plotado, sendo difícil sua visualização nas figuras.

A partir dos experimentos realizados, as seguintes métricas serão discutidas: atraso médio na entrega das mensagens e percentual de mensagens entregues.

3.2 Avaliação Experimental

As Figuras 3.1 e 3.2 apresentam o atraso médio na entrega das mensagens na rede DTN e na rede *ad hoc*, respectivamente.

Observa-se que, o atraso ocorrido na entrega das mensagens na rede DTN é muito mais elevado do que o ocorrido na rede *ad hoc*, sendo que essa diferenciação ocorre tanto em redes esparsas, quanto em redes densas. Ao verificar-se os valores de atraso apresentados com raio de 20 m (Figura 3.1a e Figura 3.2a), por exemplo, a maioria das mensagens é entregue em até 80000

²Modelo de propagação sem obstáculos [Fal09].

³Define que será utilizado um canal sem fio na comunicação entre os nodos [Fal09].

⁴Define uma antena omni-direcional para ser utilizada pelos nodos [Fal09].

⁵Implementação do protocolo IEEE 802.11 [Fal09].

segundos na rede DTN, enquanto que na rede *ad hoc*, as mensagens são entregues em até 13 segundos. Já com raio de 250 m, tem-se um atraso de até 10000 segundos na entrega da maioria das mensagens nas redes DTN (Figura 3.1e), contra um atraso de até 1 segundo nas redes *ad hoc* (Figura 3.2e).

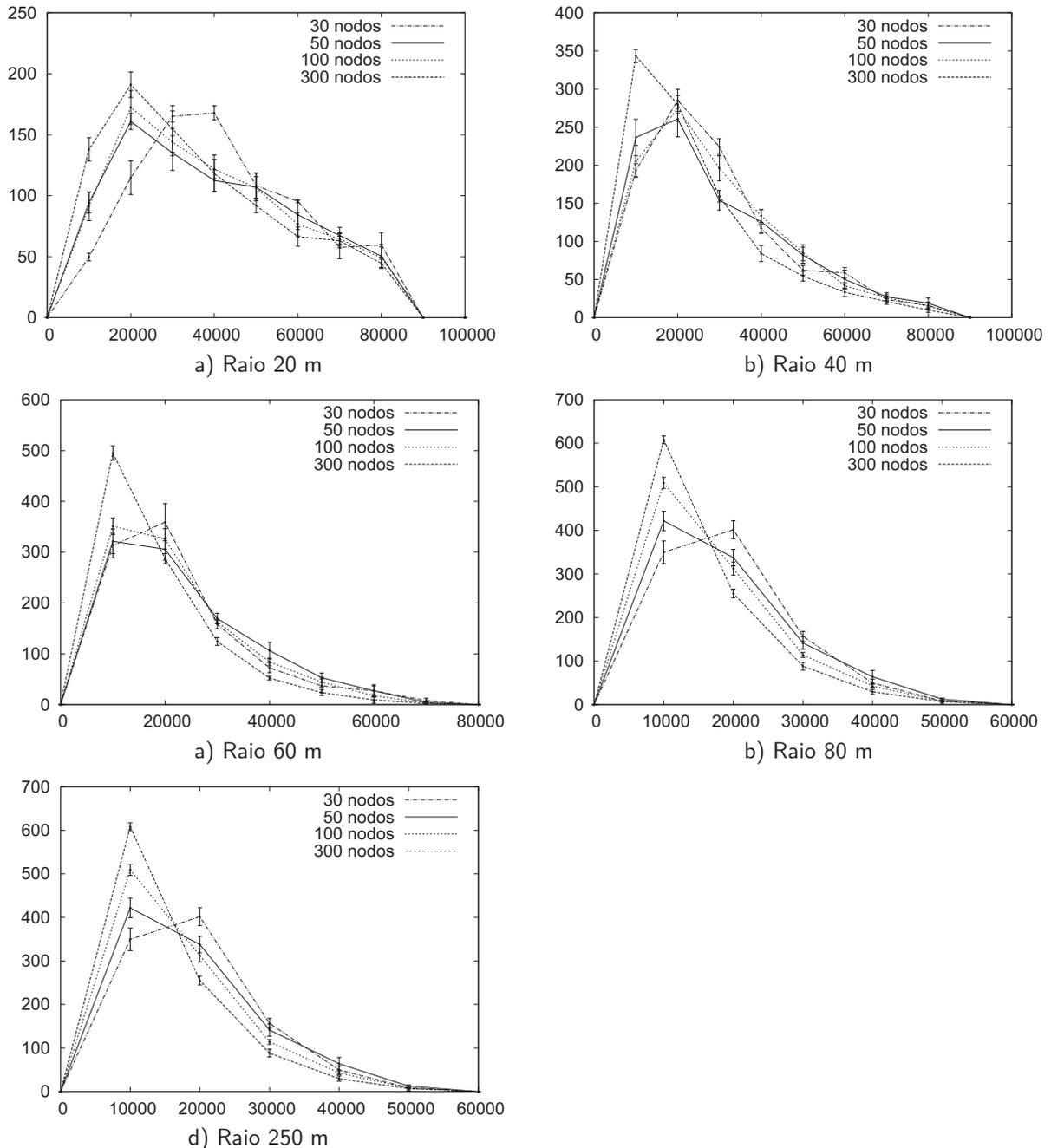


Figura 3.1: Atraso na entrega das mensagens para redes DTN. O eixo vertical representa a quantidade de mensagens entregues e o horizontal representa o atraso em segundos.

Além da diferenciação do atraso entre os dois tipos de redes, observa-se também, que na mesma rede há uma diferenciação do atraso influenciada pela densidade. Ao observar-se os valores de atraso da Figura 3.1a-e, nota-se que o atraso gerado em redes esparsas é mais elevado que o gerado em redes densas. Enquanto na Figura 3.1a as mensagens são entregues em até 80000 segundos, na

Figura 3.1c o atraso é de até 70000 segundos e na Figura 3.1e tem-se a maioria das mensagens sendo entregues em até 10000 segundos. Observando-se os valores de atraso da Figura 3.2a-e, também percebe-se as variações de atraso de acordo com a densidade.

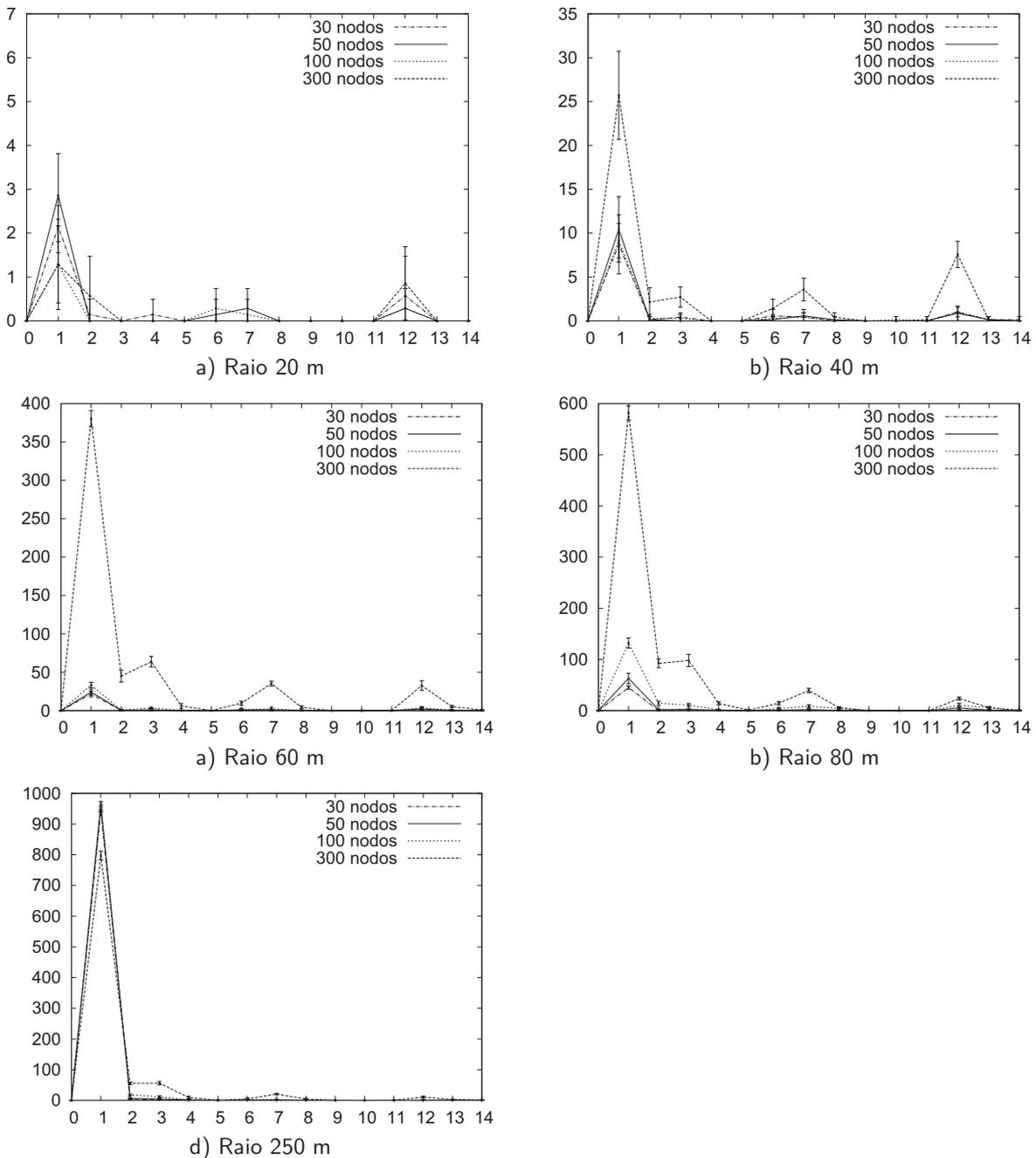


Figura 3.2: Atraso na entrega das mensagens para redes *ad hoc*. O eixo vertical representa a quantidade de mensagens entregues e o horizontal representa o atraso em segundos.

As Figuras 3.3 e 3.4 apresentam o percentual de mensagens entregues à medida que o atraso cresce para redes DTN e *ad hoc*, respectivamente. Essas figuras apresentam a mesma informação das Figuras 3.1 e 3.2, no entanto sua apresentação cumulativa favorece a análise do compromisso entre atraso e entrega.

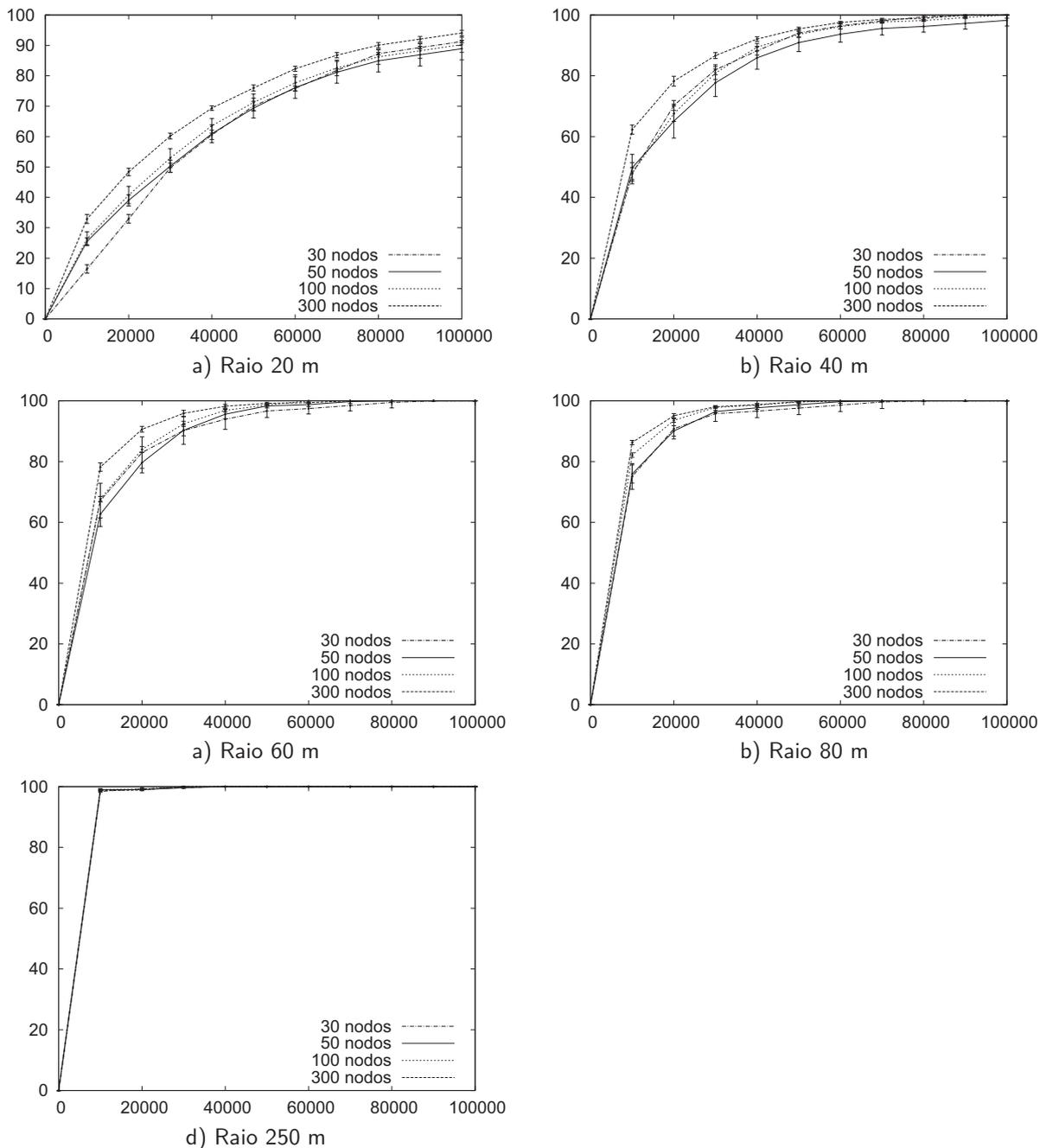


Figura 3.3: Percentual de mensagens entregues cumulativo para redes DTN. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.

Como esperado, em redes mais esparsas um número menor de mensagens é entregue, contudo esse número vai aumentando à medida que a densidade aumenta. Considerando redes de mesmo tipo, como DTN, tem-se, com raio de 20 m, até 94% das mensagens sendo entregues com um atraso de 100000 segundos (Figura 3.3a), enquanto que com raio de 250 m o percentual chega a 100% em 30000 segundos (Figura 3.3e).

A partir das Figuras 3.3 e 3.4, observa-se que, em redes esparsas, técnicas DTN podem trazer benefícios quando altos atrasos podem ser suportados. Enquanto a rede DTN com raio de alcance de 20 m e com 30 nodos entregou 91% das mensagens em 100000 segundos (Figura 3.3a), a rede

ad hoc entregou 0,3% em 11 segundos (Figura 3.4a). No entanto, a diferenciação em relação à entrega vai caindo à medida que a densidade da rede aumenta. Na rede com raio de alcance de 80 m e com 300 nodos, o percentual de mensagens entregues começa a se tornar equivalente nos dois tipos de rede, sendo que a rede *ad hoc* tem vantagem em relação à DTN devido ao baixo atraso. Essa característica pode ser observada na Figura 3.4d que apresenta um percentual de entrega de 84% com 6 segundos de atraso, enquanto a Figura 3.3d apresenta o mesmo percentual com um atraso de 10000 segundos.

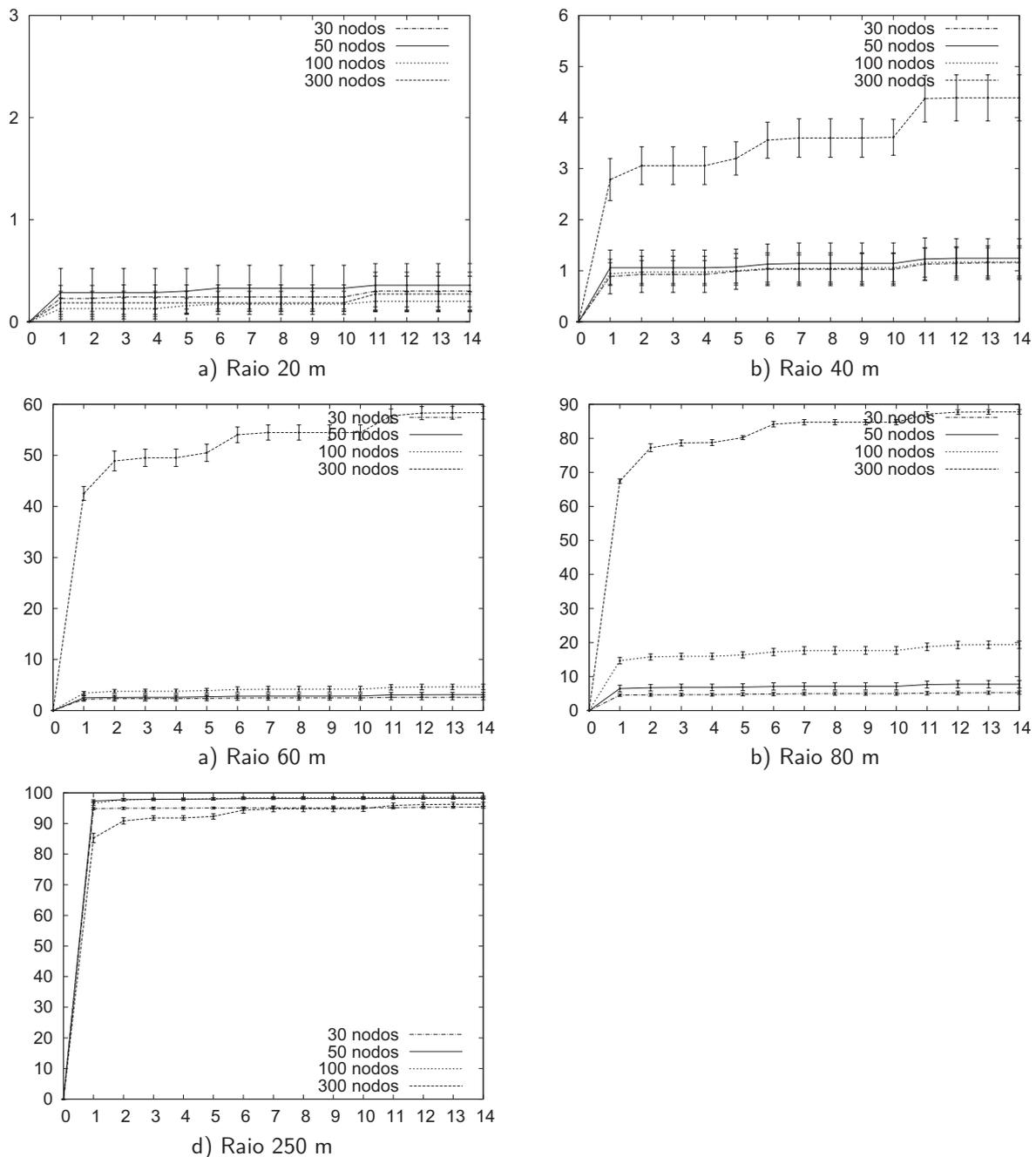


Figura 3.4: Percentual de mensagens entregues cumulativo para redes *ad hoc*. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.

Na Figura 3.5 são apresentados os dados de entrega em função do atraso, assim como na Figura 3.3, mas detalhando o percentual de mensagens entregues em até 10000 segundos. Esse detalhamento serve para melhorar a diferenciação do impacto da densidade para redes DTN e também para permitir uma comparação mais detalhada de atraso e entrega entre redes DTN e *ad hoc*.

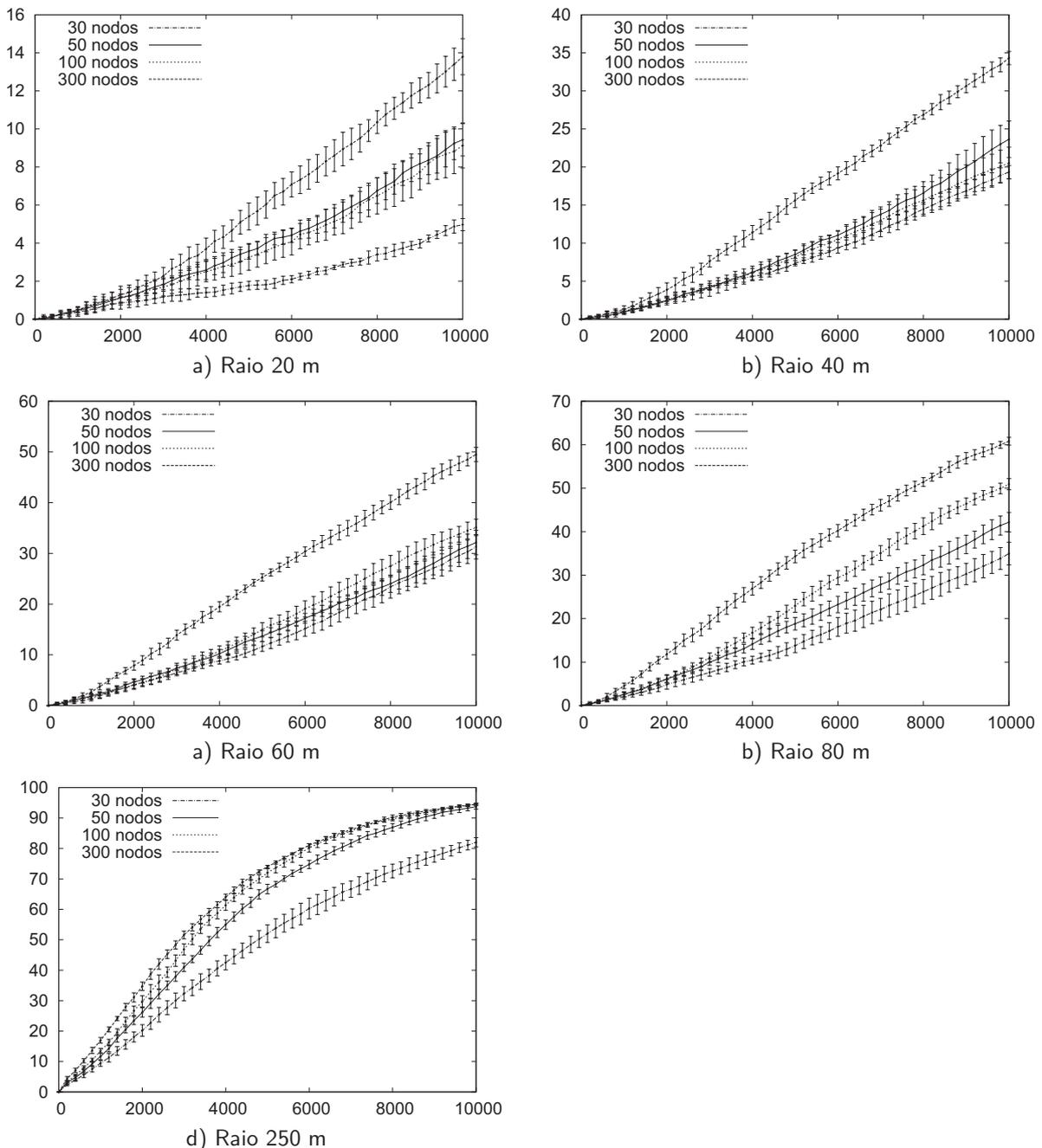


Figura 3.5: Percentual de mensagens entregues cumulativo para redes DTN em 10000 segundos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa o atraso em segundos.

Ao analisar os gráficos da Figura 3.5 para um mesmo raio, percebe-se que há uma diferenciação de atraso e entrega na alteração da densidade da rede. Essa alteração é mais perceptível ao analisar

os resultados de 30 e 300 nodos. Como exemplo, na rede com raio de 20 m (Figura 3.5a), em 10000 segundos a rede com 30 nodos entregou 5% das mensagens, enquanto a rede com 300 nodos entregou 14%. Já com raio de 250 m, nos mesmos 10000 segundos, a rede com 30 nodos entregou 82% e a com 300 nodos entregou 94%.

Em relação ao atraso entre as redes DTN (Figura 3.5) e *ad hoc* (Figura 3.4), observa-se que em nenhum momento a rede DTN possui um atraso inferior ao gerado na rede *ad hoc*, considerando um percentual de entrega compatível. Enquanto a rede *ad hoc* com raio de 20 m e 30 nodos (Figura 3.4a) entrega 0,3% das mensagens em 11 segundos, a rede DTN entrega esse mesmo percentual em 800 segundos (Figura 3.5a). Já na rede *ad hoc* com raio de 40 m e 300 nodos (Figura 3.4b) em um tempo de 6 segundos foram entregues 3,5% das mensagens, sendo que a rede DTN leva 1880 segundos para entregar esse mesmo percentual.

3.3 Considerações Finais

Os experimentos conduzidos neste Capítulo mostram um indicativo de parâmetros para o estudo de redes DTN. Quando uma rede possui densidade compatível com as configurações da Figura 3.4e, por exemplo, a utilização de estratégias de roteamento para redes DTN é questionável, pois é possível obter alto percentual de mensagens entregues e baixo atraso com estratégias para redes *ad hoc*. Já com redes de menor densidade, como a maior parte das configurações das Figuras 3.4c e 3.4d e todas configurações das Figuras 3.4a e 3.4b, a utilização de DTN é importante ou obrigatória devido aos baixos percentuais de entrega das redes *ad hoc* (0 a 7%). Isto indica que nessa faixa deve-se investir esforço de investigação.

Caso a aplicação suporte altos atrasos, as redes DTN podem ser empregadas, de forma a aumentar o percentual de mensagens entregues. Neste sentido, os gráficos da Figura 3.3 indicam quanto pode-se conseguir de entrega, utilizando tais técnicas, à medida que aumenta o atraso suportado.

4. O Protocolo APRP

Em redes DTN nodos estabelecem e perdem vizinhança com outros nodos continuamente durante a sua operação. A vizinhança é estabelecida ou perdida quando os nodos conseguem, ou respectivamente não conseguem, trocar mensagens diretamente utilizando o enlace sem fio. Como já discutido, esta variação da conectividade tem várias causas como mobilidade dos nodos, fatores ambientais, ou fatores relacionados aos nodos em questão, como por exemplo, baixos níveis de energia.

Assim, considerando que nodos carregam mensagens que devem ser roteadas com o intuito de chegarem ao seu destino, a principal decisão em um protocolo de roteamento para DTN é se uma mensagem deve ou não ser roteada de um nodo para outro quando estes estabelecem vizinhança. Essa decisão deve maximizar a taxa de entrega e minimizar o consumo de recursos de comunicação, de armazenamento, além do atraso fim-a-fim. Além disso, o desempenho de um protocolo é dependente de configurações da rede como: tipo de mobilidade dos nodos, raio de alcance considerado para as transmissões e número de nodos na área considerada, ambos impactando na densidade da rede. Neste contexto, diferentes protocolos adotam diferentes políticas para determinar se o repasse deve acontecer ou não, tal como discutido na Seção 2.4.

O protocolo APRP (*Adaptive Potential Routing Protocol*), proposto neste trabalho, visa aumentar o percentual de mensagens entregues e diminuir o número de cópias geradas na rede, reduzindo custos de comunicação, armazenamento e os impactos advindos destes. Para tanto, consideram-se redes estocásticas, assim como os protocolos citados na Seção 2.4. O APRP usa, como principal métrica para decisão de repasse das mensagens, o número de encontros em que um nodo está envolvido em um intervalo de tempo. Tal métrica é chamada de “potencial de entrega”. A estratégia de repasse assume que o comportamento de encontros de um nodo está relacionado com a história recente do mesmo. Além disso, é assumido que nodos com potencial de entrega mais alto tem uma probabilidade maior de encontrar o destino em um futuro próximo.

4.1 Cálculo do Potencial de Entrega de um Nodo

No protocolo APRP cada nodo de uma rede DTN monitora o número de encontros com outros nodos em um período de tempo, que é chamado de potencial de entrega. O potencial de entrega dos nodos altera seu valor, aumentando ou diminuindo, à medida que o número de encontros de um nodo em um período de tempo respectivamente aumenta ou diminui. A cada novo encontro de um nodo o seu potencial é recalculado registrando o novo encontro e seu momento de ocorrência, conforme o relógio local do nodo, e retirando encontros antigos da janela de tempo considerada. Assim, o potencial de entrega é uma medida histórica que considera os encontros do nodo no passado recente do mesmo.

Para o ajuste do tamanho da janela de tempo, vários experimentos foram realizados variando densidade da rede, padrão de mobilidade e tamanho da janela de tempo. A partir desses experimentos observou-se que uma janela de tempo de 4 minutos é suficiente para caracterizar diferentes níveis de potencial dos nodos da rede. Esse valor foi adotado nos experimentos realizados neste Capítulo.

As Tabelas 4.1, 4.2 e 4.3 apresentam valores médios de potenciais de entrega, obtidos por simulação, para a janela de tempo de 4 minutos, em uma área de 4500 m x 3400 m, com 30, 50, 100 e 300 nodos e para três cenários diferentes, considerando raios de 10 m, 30 m e 80 m, respectivamente. Além disso, foi utilizado o padrão de mobilidade *Shortest Path Map Based Movement*, descrito na Seção 2.3, e dois tipos de nodos foram simulados: pessoas e automóveis. Os demais parâmetros utilizados nas simulações estão descritos na Seção 4.4.

Tabela 4.1: Valor médio do potencial para raio de 10 m.

	30 nodos	50 nodos	100 nodos	300 nodos
Pessoas	1,8	2,3	3,6	8,4
Automóveis	2,3	3,1	5,1	13,11

Tabela 4.2: Valor médio do potencial para raio de 30 m.

	30 nodos	50 nodos	100 nodos	300 nodos
Pessoas	1,9	2,4	3,9	9,6
Automóveis	2,4	3,4	5,8	15,34

Tabela 4.3: Valor médio do potencial para raio de 80 m.

	30 nodos	50 nodos	100 nodos	300 nodos
Pessoas	2,1	2,8	4,7	11,9
Automóveis	2,9	4,2	7,4	20,34

Analisando os resultados das Tabelas 4.1, 4.2 e 4.3 percebe-se que os nodos com maior velocidade, os automóveis nesse caso, possuem um potencial mais elevado que aqueles com menor velocidade, como as pessoas. Um outro parâmetro que altera o potencial é a densidade da rede, impactada pelo raio de alcance e a quantidade de nodos na área simulada. Os nodos das redes mais densas têm potenciais mais elevados, enquanto os nodos das redes mais esparsas ou com raio menor têm potenciais mais baixos. Os resultados indicam que o potencial dos nodos está representando uma medida coerente dos encontros ocorridos com eles, podendo ser utilizado de forma comparativa. Conforme experimentos realizados com outros padrões de mobilidade, como o *Random Waypoint* e o *Map Based Movement*, descritos na Seção 2.3, observa-se que o comportamento do potencial se repete para as mesmas configurações descritas anteriormente.

4.2 Roteamento no APRP

A decisão de roteamento em uma rede DTN acontece a cada encontro de dois nodos. Cada nodo do par formado no encontro pode tanto enviar como receber mensagens do outro nodo. A descrição a seguir se dá em somente um sentido do par formado, pois o comportamento é simétrico.

Um nodo busca nodos que tenham uma diferença de potencial, com relação ao seu, que justifique o repasse de uma mensagem a ele, pois o repasse gasta recursos de comunicação e armazenamento, além de uma cópia da mensagem. A diferença de potencial que justifica o repasse é relativa ao maior valor de potencial conhecido. Devido à natureza da rede em discussão, essa estimativa é local, e não global, e dependente dos encontros ocorridos por um nodo. De maneira mais detalhada, o valor de maior potencial é identificado por cada nodo, a cada encontro. Assim, cada nodo dispõe do seu valor de potencial e de um valor máximo de potencial conhecido por ele.

A decisão de repasse depende da existência de mensagens para repasse e dos valores de potencial e potencial máximo conhecido de cada nodo. Assim, o repasse de uma mensagem de um nodo origem o para um nodo encontrado n ocorre se:

$$P_n > P_o + \delta(P_{max-o} - P_o) \quad (4.1)$$

Onde P_o é o potencial de entrega do nodo origem; P_n é o potencial de entrega do nodo encontrado n ; P_{max-o} é o potencial máximo conhecido por o ; e δ é um valor de ajuste para o diferencial de potencial desejado. Experimentos¹ com variados valores de δ foram realizados para identificar o limite que forneça uma boa relação entre taxa de entrega, atraso e *overhead* gerado na rede. Os valores de δ testados variaram de 0,2 a 0,6. Identificou-se que à medida que o valor de δ diminui ambos *overhead* e a taxa de entrega aumentam, sendo que o atraso é reduzido. O valor de δ utilizado nos experimentos está apresentado na Seção 4.4, bem como resultados que mostram o comportamento das métricas para dois valores de δ .

Complementarmente, o APRP limita o número de cópias de uma mensagem. A estratégia usada é semelhante a do protocolo *Spray and Wait* [Spy08] com *spraying* binário, ou seja, a cada encontro um nodo pode repassar créditos de mensagens ao nodo encontrado. Tanto o nodo originador como o encontrado mantém cópias da mensagem para futura tentativa de repasse ou entrega direta. Assim, a origem de uma mensagem inicia com um crédito de C cópias; quando o repasse de uma mensagem é feito, ela delega $\lfloor C/2 \rfloor$ dos créditos para o outro nodo e mantém $\lceil C/2 \rceil$ para si; desde que hajam $C > 1$ cópias da mensagem, caso contrário a transmissão direta é utilizada. Caso não haja espaço em *buffer* para armazenar a mensagem, a estratégia FIFO é utilizada para descarte.

Assim como em outros protocolos de roteamento, como Epidêmico, PRoPHET e *Spray and Wait* por exemplo, os nodos possuem um *summary vector* com a identificação de todas as mensagens armazenadas por eles. Quando dois nodos se encontram, eles trocam o *summary vector* e verificam quais mensagens são novas e precisam ser repassadas. Dessa forma, evita-se utilizar recursos da rede com o repasse de mensagens já armazenadas pelos nodos.

¹Os experimentos utilizaram a configuração descrita na Seção 4.4.

4.3 Algoritmo

O Algoritmo 4.1 apresenta o mecanismo de roteamento usado no APRP. Tal protocolo possui os seguintes parâmetros que devem ser configurados:

- δ : valor de ajuste para o diferencial de potencial desejado;
- C : número de créditos que uma mensagem possui;
- W_p : tamanho da janela de tempo usada na atualização da *contact_list_p*.

O nodo que origina uma mensagem ou que possui mensagens a serem repassadas possui as seguintes estruturas e variáveis:

- o : identificação do próprio nodo;
- P_o : potencial de entrega do próprio nodo;
- n : identificação do nodo encontrado. Essa informação é obtida do nodo encontrado no momento do contato entre os nodos;
- P_n : potencial de entrega do nodo encontrado. Essa informação é obtida do nodo encontrado no momento do contato entre os nodos;
- P_{max-o} : potencial máximo conhecido pelo nodo origem. Calculado pelo nodo assim que o potencial de entrega do nodo encontrado é obtido;
- $time$: tempo em que houve o contato com o nodo encontrado;
- *contact_list*: lista de contatos que informa quem são os vizinhos do próprio nodo, no formato $\langle n, P_n, time \rangle$. Essa lista é utilizada no cálculo do potencial de entrega dos nodos. Cada entrada da lista fica válida por um tempo, até que a janela de tempo (W_p) expire. Um vizinho é definido como um outro nodo que está em alcance direto no raio de alcance da rede sem fio usada;
- m : identificação da mensagem a ser repassada ou entregue ao destino final. Todas as mensagens possuem uma identificação, necessária para diferenciar mensagens antigas de mensagens novas, mensagens duplicadas e mensagens que devem ser retiradas do *buffer*;
- *summary_vector_o*: lista com a identificação das mensagens e o número de créditos que estão sendo armazenados pelo próprio nodo, no formato $\langle m, C \rangle$;
- *summary_vector_n*: lista com a identificação das mensagens que estão sendo armazenadas pelo nodo encontrado.

 Algoritmo 4.1: Mecanismo de roteamento do APRP.

```

procedure nodeMeetingNeighbor()
  obtain(n, Pn, summary_vector_n, time)
  updateContactList(contact_list, n, Pn, time, Wp)
  if ( $P_n > P_{max-o}$ ) then
     $P_{max-o} = P_n$ 
  end if
  calculatePo(contact_list, Po)
  if (nodeHasMessage(summary_vector_o)) then
    directDelivery(summary_vector_o, n, summary_vector_n)
    if ( $P_n \geq (P_o + \delta * (P_{max-o} - P_o))$ ) then
      sendMessageToNeighbor(summary_vector_o, n, summary_vector_n)
    end if
  end if

procedure directDelivery(summary_vector_o, n, summary_vector_n)
  for all m in summary_vector_o do
    if ((destination(m) = n) and (!neighborHasMessage(m, summary_vector_n))) then
      sendDirectMessage(m, n)
      removeMessageList(summary_vector_o, m)
    end if
  end for

procedure sendMessageToNeighbor(summary_vector_o, n, summary_vector_n)

  for all  $\langle m, C \rangle$  in summary_vector_o do
    if (( $C > 1$ ) and (!neighborHasMessage(m, summary_vector_n))) then
      sendMessage(n, m,  $\lfloor C/2 \rfloor$ )
       $C = \lceil C/2 \rceil$ 
      updateMsgList(summary_vector_o, m, C)
    end if
  end for

procedure calculatePo(contact_list, Po) {Calcula o potencial do nodo origem.}
procedure createNewMessage(m, summary_vector_o, C) {Inclui a identificação da nova mensagem no
summary vector do nodo e configura o número de créditos para a mensagem criada.}
procedure neighborHasMessage(m, summary_vector_n) {Verifica se a mensagem m está presente no
summary vector do nodo encontrado. Caso esteja retorna true, caso contrário retorna false.}
procedure nodeHasMessage(summary_vector_o) {Verifica se há mensagens para serem entregues no
summary vector no próprio nodo, retornando true ou false.}
procedure obtain(n, Pn, summary_vector_n, time) {Retorna a identificação, o potencial e o summary vector
do nodo encontrado, bem como o tempo do encontro.}
procedure removeMessageList(summary_vector_o, m) {Remove mensagem da lista de mensagens.}
procedure sendMessage(n, m, C) {Repassa mensagem para o nodo.}
procedure sendDirectMessage(m, n) {Envia mensagem para o destino.}
procedure updateContactList(contact_list, n, Pn, time, Wp) {Inclui a identificação do nodo encontrado na
lista de contatos e remove contatos antigos de acordo com a janela de tempo.}
procedure updateMsgList(summary_vector_o, m, C) {Atualiza a lista de mensagens dando o novo crédito
para a mensagem repassada.}
  
```

Quando uma mensagem é criada, o procedimento *createNewMessage*($m, summary_vector_o, C$) é executado. Tal procedimento inclui a identificação da mensagem criada no *summary vector* do nodo e configura o número de créditos com o valor recebido por parâmetro.

Ao encontrar um nodo n , o procedimento *nodeMeetingNeighbor*() é executado. A identificação, o potencial e a lista com a identificação das mensagens mantidas pelo nodo encontrado, bem como o tempo do encontro são obtidos pelo procedimento *obtain*($n, P_n, summary_vector_n$). O nodo encontrado é adicionado na lista de contatos e os contatos antigos são removidos da mesma de acordo com a janela de tempo W_p . O potencial máximo conhecido pelo nodo é atualizado e o procedimento *calculateP_o*(*contact_list*, P_o) calcula o novo potencial de entrega do nodo origem. Caso haja mensagens para serem repassadas, o procedimento *directDelivery*(*summary_vector_o*, $n, summary_vector_n$) identifica se o nodo encontrado é destino de alguma mensagem presente em sua lista para que seja feita a entrega direta e a remoção da mesma. A entrega direta somente será realizada caso o nodo encontrado ainda não possua a mensagem a ser entregue. O repasse das demais mensagens é realizado se o potencial do nodo encontrado for maior ou igual ao valor que justifica o repasse (descrito na Seção 4.2). O envio da mensagem e o ajuste dos créditos é realizado pelo procedimento *sendMessageToNeighbor*(*summary_vector_o*, $n, summary_vector_n$), o qual transmite $\lfloor C/2 \rfloor$ dos créditos para o nodo encontrado e atualiza os créditos locais da mensagem em $\lfloor C/2 \rfloor$. O repasse somente será realizado caso o nodo encontrado ainda não possua a mensagem a ser entregue.

4.4 Avaliação Experimental

4.4.1 Parâmetros da Simulação

A avaliação do APRP foi realizada através de implementação utilizando o simulador ONE (*Opportunistic Network Environment*) [Ker07], um ambiente de simulação de eventos discretos específico para DTN.

Protocolos Comparados

Através da implementação do protocolo APRP foi possível comparar o seu comportamento com os resultados obtidos de estratégias de roteamento já existentes na literatura. A comparação foi realizada com os protocolos Epidêmico, *Spray and Wait* e P_RoPHET, os quais já vêm implementados no simulador ONE. Os dois primeiros foram selecionados pois ambos são restrições da estratégia de *flooding*, assim como o APRP, sendo que o *Spray and Wait* é utilizado como base do protocolo proposto. Já o P_RoPHET utiliza informações históricas acerca do comportamento dos nodos para a tomada de decisão sobre o repasse das mensagens, sendo o mais referenciado na literatura com essa característica. Além disso, tal protocolo está em fase de padronização pelo IETF [Lin09]. Para este último protocolo foi necessário configurar o parâmetro de transitividade, o qual foi configurado com 0,25 (sugerido em [Lin03b]).

Parametrização do APRP

O protocolo APRP precisa ter configurados três parâmetros: o tamanho da janela de tempo onde os encontros são válidos para o cálculo do potencial de entrega (W_p), o δ da Equação 4.1 e o limite (máximo) no número de vezes que uma mensagem pode ser copiada na rede (C). A seguir estão indicados os valores que foram armazenados em cada um dos parâmetros.

- Janela de Tempo (W_p): conforme comentado na Seção 4.1, foram realizados vários experimentos variando o tamanho da janela de 2 a 16 minutos e foi observado que uma janela de tempo de 4 minutos é suficiente para caracterizar diferentes níveis de potencial nos nodos da rede;
- δ : igualmente, experimentos com valores de δ entre 0,2 e 0,6 foram realizados. Os resultados apresentados nesta Seção foram realizados com δ de 0,2, pois observou-se uma melhor relação entre percentual de mensagens entregues, atraso e *overhead*- vide Seção 4.4.5;
- C : este é um parâmetro utilizado tanto no APRP quanto no *Spray and Wait* e foi configurado em 6 em ambos protocolos (sugerido em [Ker07]). Esse valor permite comparação com resultados da literatura como em [Spy08].

Cenários de Simulação

- Área Simulada: a área simulada tem um tamanho de 4500 m x 3400 m e utiliza um mapa que faz parte da área central da cidade de Helsinki, na Finlândia. Este mapa vem com o simulador, sendo utilizado pelos padrões de mobilidade *Shortest Path Map Based Movement* e *Map Based Movement* [Ker07];
- Número de Nodos: o número de nodos varia entre 30, 50, 100 e 300 nodos;
- Tipos de Nodos: considerando o número de nodos, metade representa pessoas que carregam dispositivos móveis e a outra metade representa automóveis;
- Velocidade dos Nodos: os automóveis diferem das pessoas basicamente em velocidade. As pessoas assumem velocidades que variam entre [0,5, 1,0] m/s e os automóveis assumem velocidades entre [2,7, 13,9] m/s;
- Movimento dos Nodos - padrão de mobilidade: os nodos se movem de acordo com os padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, descritos na Seção 2.3;
- Recursos dos Nodos: os automóveis possuem um tamanho de *buffer* de 40 Mbytes, enquanto os dispositivos carregados pelas pessoas possuem um tamanho de *buffer* de 20 Mbytes;
- Raio de Transmissão dos Nodos: a distância alcançada pelo sinal dos nodos varia em 10 m, 30 m e 80 m. Esta é uma outra forma de alterar a densidade da rede;

- Taxa de Transmissão: as mensagens são transmitidas a uma velocidade de 2 Mbps;
- Número de Mensagens: 5000 mensagens são criadas de origens randômicas para pontos destino randômicos na rede. O TTL (*Time to Live*) das mensagens é considerado infinito;
- Tamanho das Mensagens: o tamanho das mensagens varia de 100 Kbytes a 2 Mbytes;
- Tempo de Simulação: os experimentos rodam por aproximadamente 55 horas simuladas. Este valor foi definido através de um estudo para tornar o resultado da simulação o mais exato possível. Conforme observado no Anexo A, através de métodos analíticos pode-se modelar padrões de mobilidade para obter a distribuição espacial de nodos (distribuição da probabilidade de um nodo estar em uma posição da área analisada). Portanto, foi ajustado o tempo de simulação para atingir a distribuição espacial do nodo compatível com a obtida através de método analítico para o padrão de mobilidade *Random Waypoint*.

Para cada cenário simulado, foram realizadas sete rodadas de simulação. Os resultados apresentados nesta seção foram obtidos pela média de todas as rodadas. Para cada valor apresentado, as barras verticais associadas representam um intervalo de confiança de 95%. Contudo, algumas vezes o intervalo de confiança está bastante próximo do valor plotado, sendo difícil sua visualização nas figuras.

A seguir estão apresentados os resultados para três métricas descritas na Seção 2.5: percentual de mensagens entregues, *overhead* e atraso médio na entrega das mensagens. As tabelas presentes na Seção 4.4.7 apresentam os valores numéricos de todas as figuras expostas a seguir.

4.4.2 Percentual de Mensagens Entregues

As Figuras 4.1, 4.2 e 4.3 apresentam o percentual de mensagens entregues para o cenário descrito anteriormente, onde os nodos se movimentam de acordo com os padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente.

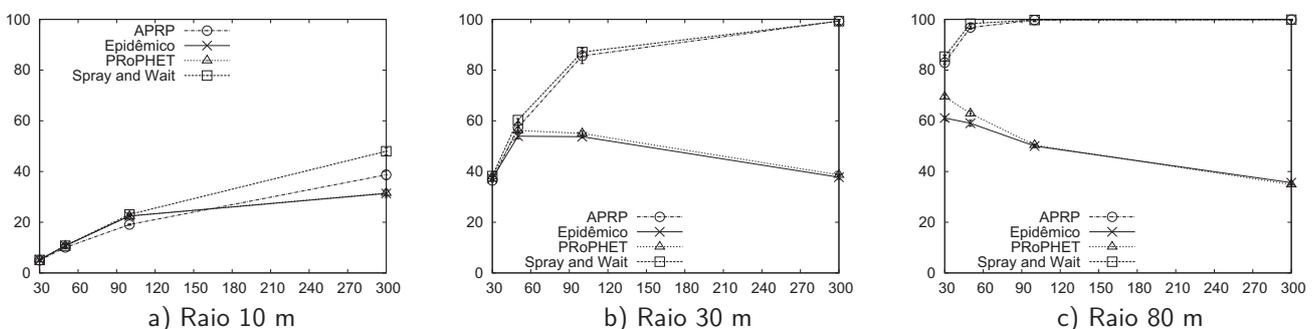


Figura 4.1: Percentual de mensagens entregues com *Random Waypoint*. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.4 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

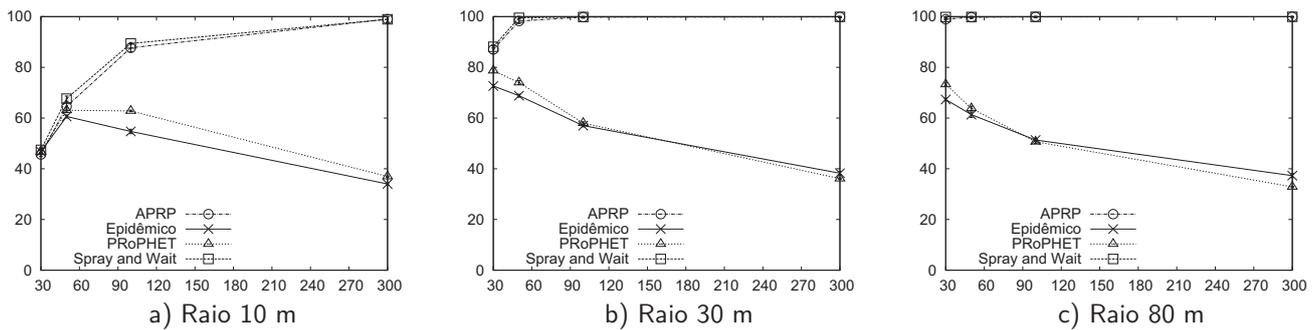


Figura 4.2: Percentual de mensagens entregues com *Shortest Path Map Based Movement*. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.5 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

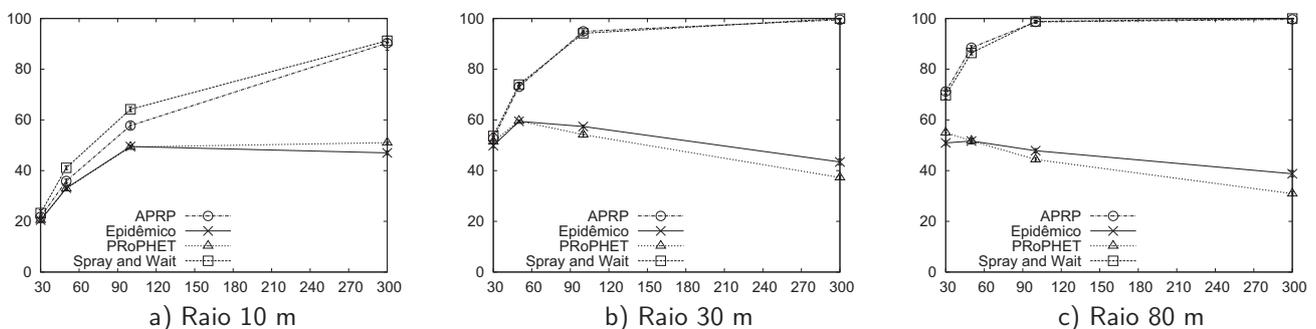


Figura 4.3: Percentual de mensagens entregues com *Map Based Movement*. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.6 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

Ao observar os valores apresentados nas Figuras 4.1, 4.2 e 4.3 percebe-se que o percentual de mensagens entregues para os protocolos Epidêmico e PRoPHET é próximo em todos os cenários simulados e vai diminuindo conforme a quantidade de nodos na rede aumenta, com exceção do experimento com Raio de alcance de 10 m e padrão de mobilidade *Random Waypoint*, onde houve um pequeno aumento. A diminuição ocorre porque há uma maior quantidade de mensagens recebidas pelos nodos, que faz com que os *buffers* fiquem cheios e ocorra uma maior perda de mensagens. Uma análise semelhante foi realizada em [Lin09], onde foram observados valores bastante próximos entre os percentuais de mensagens entregues desses dois protocolos.

Já o percentual de mensagens entregues do protocolo APRP é superior aos percentuais dos protocolos Epidêmico e PRoPHET na maioria dos casos. Como APRP tem a característica de limitar o número de cópias das mensagens, não irá ocorrer sobrecarga nos *buffers* da forma como ocorre com os protocolos Epidêmico e PRoPHET, mesmo quando a densidade da rede aumenta. Essa característica também ocorre com o protocolo *Spray and Wait*, o que torna os percentuais de mensagens entregues dos dois protocolos, APRP e *Spray and Wait*, bastante próximos. Como um exemplo, para os resultados apresentados na Figura 4.3b, a diferença entre percentual de mensagens entregues dos protocolos APRP e *Spray and Wait* ficou em média em 0,3%.

Uma outra característica que pode-se observar é a proximidade dos valores do percentual de mensagens entregues para os quatro protocolos quando a rede é muito esparsa, com 30 nodos, por exemplo. Nesses casos, os nodos encontram poucos nodos, o que acarreta em uma entrega direta das mensagens por todos os protocolos.

Além disso, observa-se que o padrão de mobilidade tem influência no percentual de mensagens entregues, como observado em [Cam02] e [Jar03], sendo que o percentual é maior com o padrão de mobilidade *Shortest Path Map Based Movement*, considerando os mesmos cenários.

4.4.3 Overhead

As Figuras 4.4, 4.5 e 4.6 apresentam o *overhead* gerado pelos protocolos à medida que a quantidade de nodos na rede aumenta. Os dados apresentados nessas figuras são relativos aos três padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*.

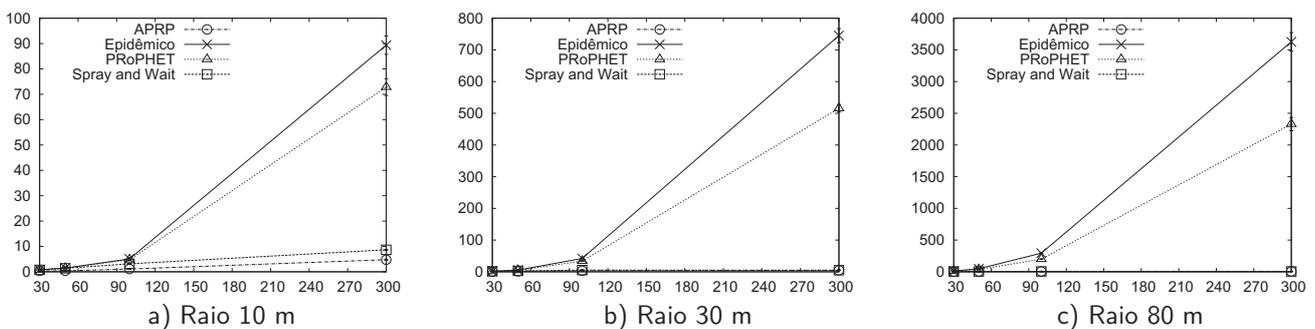


Figura 4.4: *Overhead* gerado na rede com *Random Waypoint*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.7 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

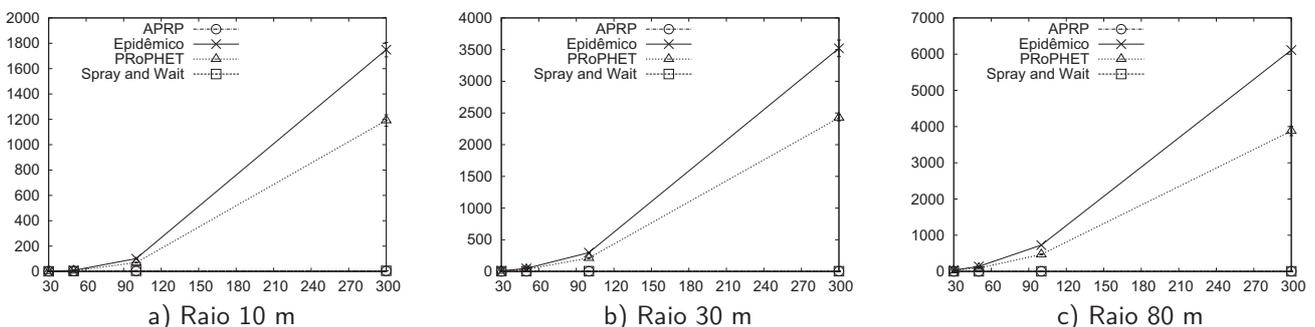


Figura 4.5: *Overhead* gerado na rede com *Shortest Path Map Based Movement*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.8 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

Em todos os experimentos realizados o *overhead* do protocolo APRP ficou inferior principalmente em relação aos protocolos Epidêmico e PRoPHET. Esse é um resultado muito importante pois indica

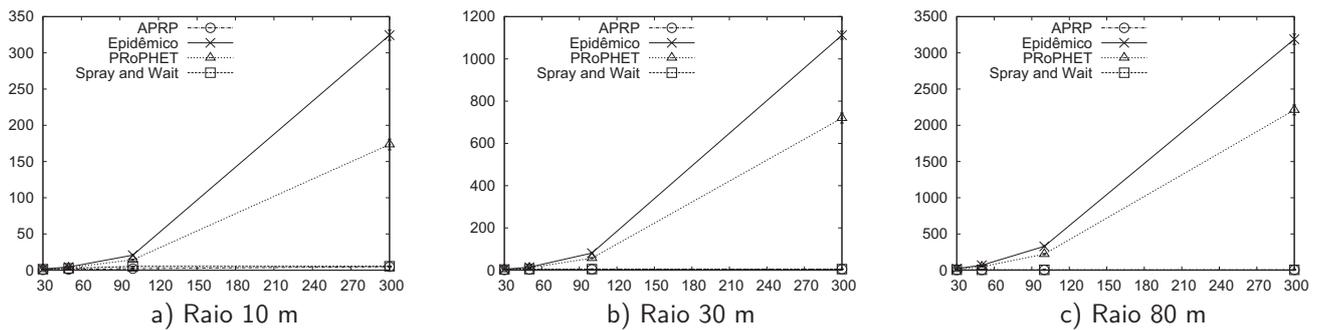


Figura 4.6: *Overhead* gerado na rede com *Map Based Movement*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.9 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

que menos recursos dos nodos estão sendo usados. Além disso, observou-se que, ao se aumentar a quantidade de nodos na rede e o raio, o *overhead* gerado pelos protocolos também aumenta. Esse aumento é maior nos protocolos Epidêmico e PRoPHET, pois ambos não impõem um limite superior no número de cópias para as mensagens. Como um exemplo, para os resultados apresentados na Figura 4.6b, o APRP reduziu o *overhead* em média 85,64% e 81,94%, em comparação aos protocolos Epidêmico e PRoPHET, respectivamente.

Ressalta-se que os valores de *overhead* para os protocolos APRP e *Spray and Wait* são baixos e, devido à escala usada nas Figuras 4.4, 4.5 e 4.6, os mesmos estão pouco visíveis. Portanto, as Figuras 4.7, 4.8 e 4.9 apresentam a diferença do *overhead* entre as duas estratégias para os padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente. Observa-se que com o aumento da quantidade de nodos, um maior número de encontros ocorre, o que ocasiona um maior encaminhamento de cópias das mensagens. Contudo, o *overhead* gerado pelo APRP se mostrou inferior ao *Spray and Wait* em todos os experimentos realizados. Como um exemplo, para os resultados apresentados na Figura 4.9b, o APRP reduziu o *overhead* em média 30,51% em comparação ao *Spray and Wait*.

Além disso, observa-se que os valores do *overhead* gerado pelos protocolos não é o mesmo para os três padrões de mobilidade. Contudo, a diferença entre eles não é tão grande quanto a diferença que ocorre no percentual de mensagens entregues. Dessa forma, verifica-se que essa métrica não é tão influenciada pelo padrão de mobilidade utilizado.

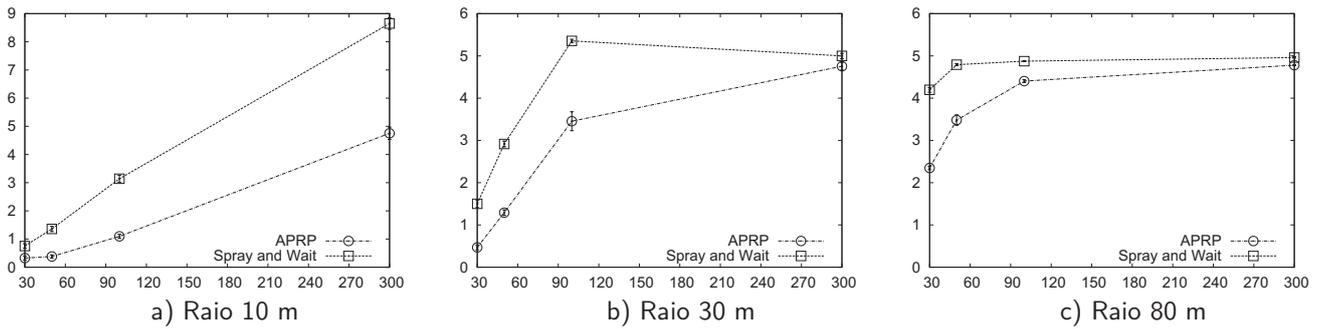


Figura 4.7: *Overhead* dos protocolos APRP e *Spray and Wait* com *Random Waypoint*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.7 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

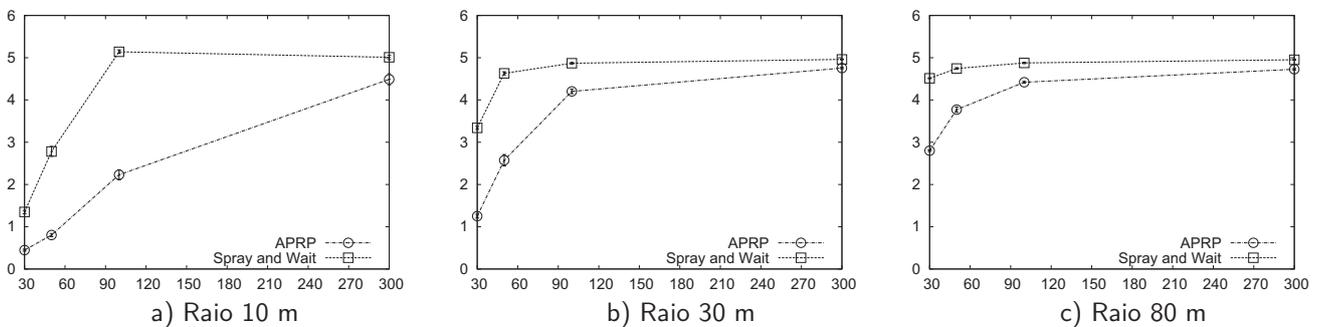


Figura 4.8: *Overhead* dos protocolos APRP e *Spray and Wait* com *Shortest Path Map Based Movement*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.8 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

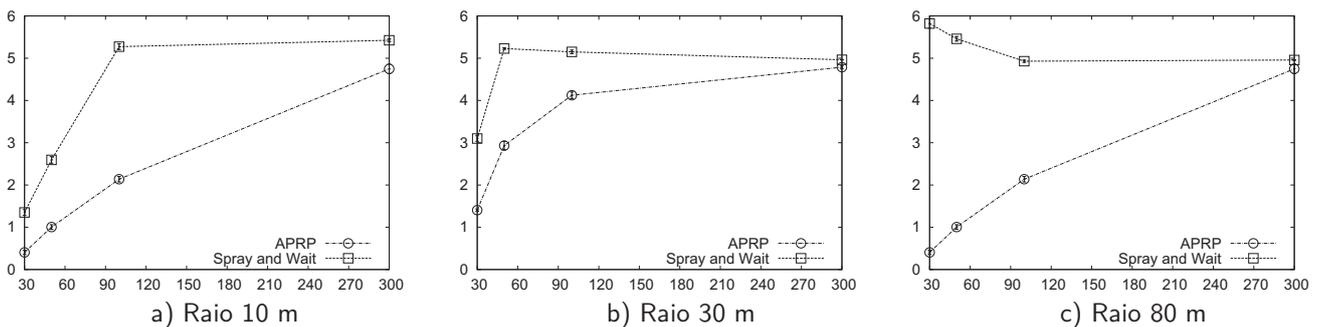


Figura 4.9: *Overhead* dos protocolos APRP e *Spray and Wait* com *Map Based Movement*. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 4.9 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

4.4.4 Atraso na Entrega das Mensagens

Outro parâmetro investigado foi o atraso médio na entrega das mensagens. As Figuras 4.10, 4.11 e 4.12 apresentam os valores obtidos dos experimentos realizados para os três padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente.

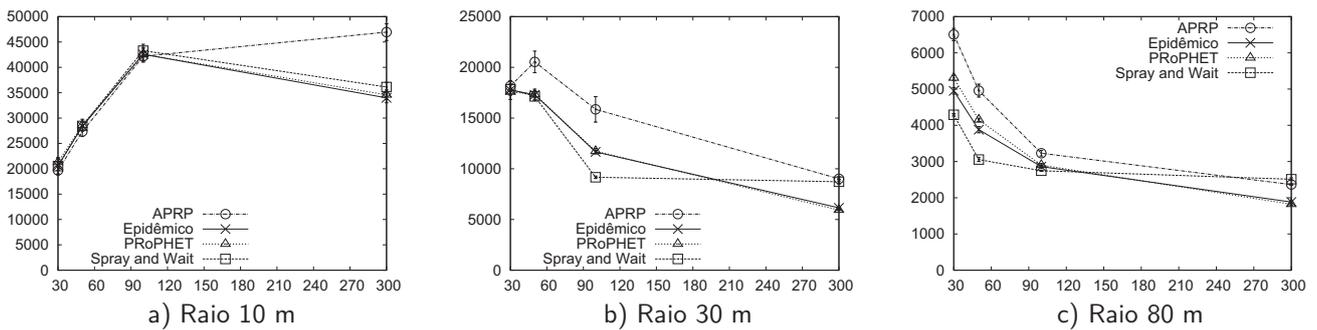


Figura 4.10: Atraso na entrega das mensagens com *Random Waypoint*. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.10 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

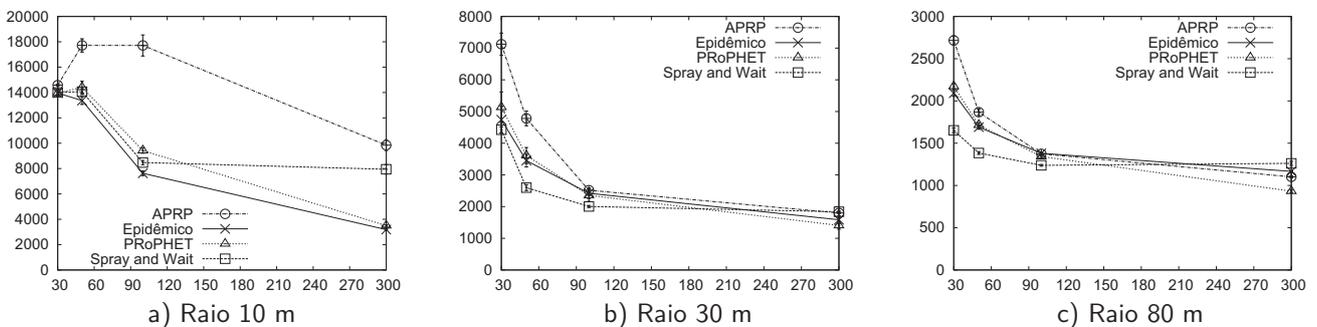


Figura 4.11: Atraso na entrega das mensagens com *Shortest Path Map Based Movement*. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.11 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

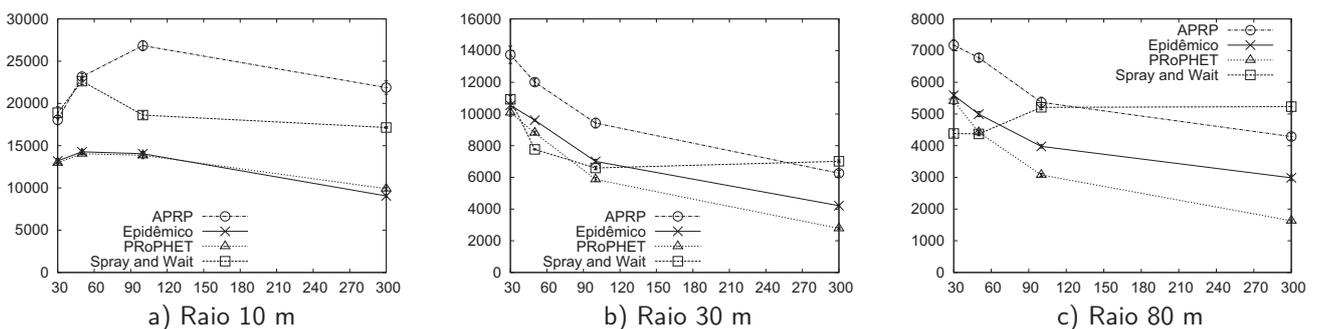


Figura 4.12: Atraso na entrega das mensagens com *Map Based Movement*. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 4.12 apresenta os valores numéricos dessa figura. Os valores correspondentes ao APRP estão representados como APRP delta 0,2.

Observa-se que o atraso no protocolo APRP é mais elevado que nos demais protocolos. Contudo, geralmente essa diferença vai diminuindo à medida que a quantidade de nodos na rede aumenta, assim como o raio. Essa característica ocorre pois no protocolo APRP uma quantidade menor de nodos possui cópia das mensagens (como observado anteriormente), uma vez que o repasse é

realizado para nodos de potencial mais elevado, ocasionando um atraso maior para encontrar o destinatário. Como um exemplo, para os resultados apresentados na Figura 4.12b, o APRP gerou um aumento no *atraso* em média de 34,57%, 64,58% e 28,15%, em comparação aos protocolos Epidêmico, PRoPHET e *Spray and Wait*, respectivamente.

Além disso, observa-se que em cenários bastante esparsos, com 30 nodos e raio de alcance de 10 m por exemplo, os valores de atraso são bastante próximos, pois uma quantidade pequena de encontros ocorre, sendo realizada a entrega direta das mensagens, conforme comentado na Seção 4.4.2.

4.4.5 Análise do Delta

O δ (delta) é um parâmetro que influencia na decisão de repasse realizada pelo protocolo APRP. Experimentos com valores de δ entre 0,2 e 0,6 foram realizados para identificar o valor que traria um melhor desempenho ao protocolo proposto. As Figuras 4.13, 4.14 e 4.15 apresentam as métricas de percentual de mensagens entregues, *overhead* e atraso médio na entrega das mensagens, respectivamente, para δ configurado com 0,2 e 0,5 e para o padrão de mobilidade *Map Based Movement*. Como o comportamento das métricas para os demais padrões de mobilidade apresentou-se semelhante com a variação do δ , resolveu-se apresentar apenas um dos padrões para realizar a análise.

A partir dos resultados plotados observa-se que o valor do δ tem influência nas métricas de desempenho. À medida que o valor de δ diminui, o atraso médio na entrega das mensagens também diminui, sendo que *overhead* e o percentual de mensagens entregues aumentam. Devido aos mais baixos valores de atraso, o δ com 0,2 foi utilizado nos experimentos realizados neste Capítulo.

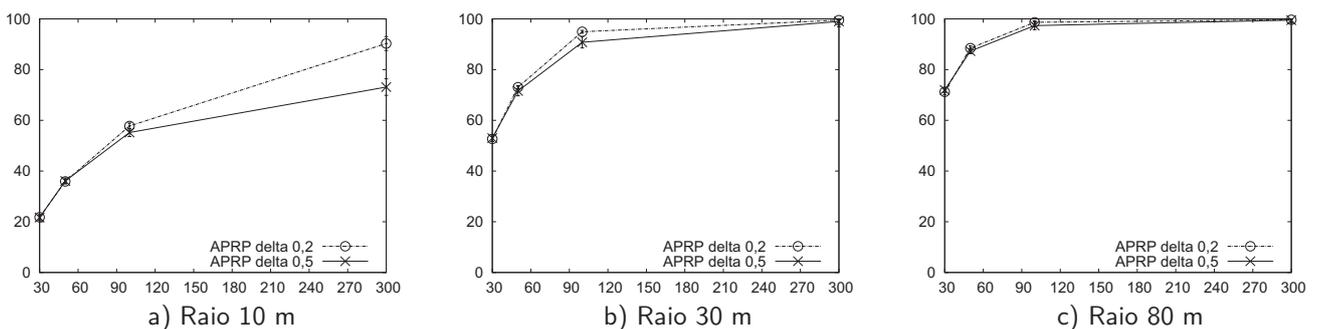


Figura 4.13: Percentual de mensagens entregues. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 4.6 apresenta os valores numéricos dessa figura.

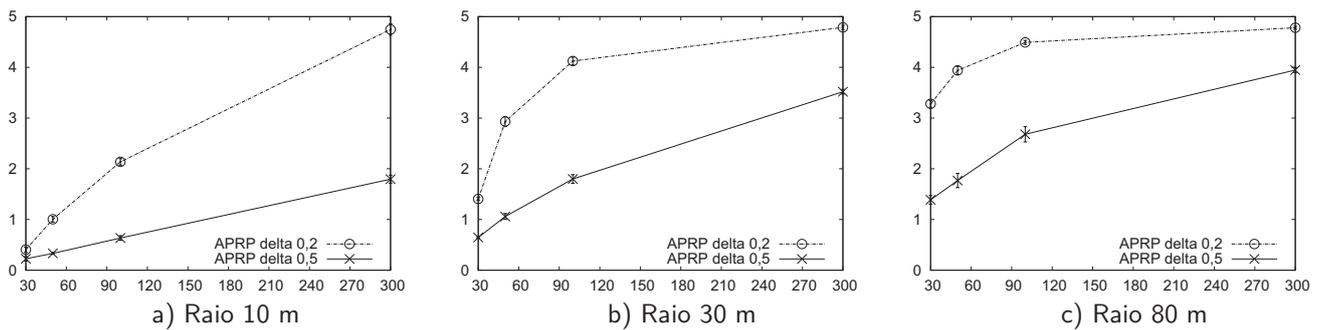


Figura 4.14: *Overhead* gerado na rede. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nós. A Tabela 4.9 apresenta os valores numéricos dessa figura.

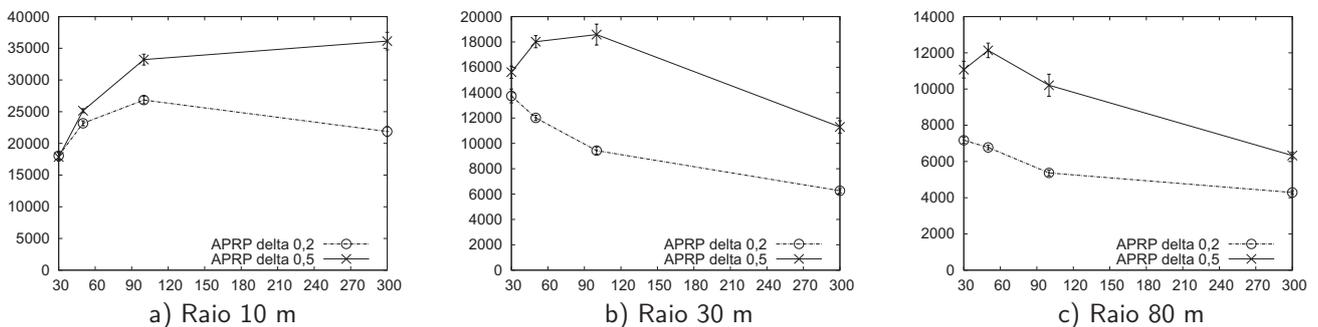


Figura 4.15: Atraso na entrega das mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nós. A Tabela 4.12 apresenta os valores numéricos dessa figura.

4.4.6 Avaliação da Sobrecarga nos Nós

Uma análise importante a ser feita é o impacto da estratégia proposta para os nós de maior potencial, uma vez que estes tendem a receber mais mensagens. A análise foi realizada considerando as mensagens recebidas pelos dois grupos de nós utilizados nos experimentos: automóveis e pessoas. Conforme apresentado na seção 4.1, os automóveis possuem um valor de potencial maior, então pretende-se avaliar a sobrecarga nesse grupo.

Para tanto, calculou-se a média das mensagens recebidas pelos dois grupos, bem como para os dois valores de δ (0,2 e 0,5) apresentados anteriormente. Os resultados obtidos foram comparados com os resultados do protocolo *Spray and Wait*. As Figuras 4.16 e 4.17 mostram os valores da média das mensagens recebidas pelos dois grupos para o padrão de mobilidade *Map Based Movement*. Na Figura 4.16 está sendo feita a comparação do APRP com δ configurado em 0,2 e na Figura 4.17 em 0,5. Os gráficos das figuras estão considerando todas as mensagens recebidas pelos nós, inclusive aquelas que foram descartadas por falta de espaço em *buffer*.

O APRP com δ em 0,2 sobrecarrega mais os nós da rede do que com δ em 0,5 e, consequentemente, possui um maior *overhead*.

Analisando os resultados apresentados na Figura 4.16, percebe-se que ambos protocolos (APRP e *Spray and Wait*) possuem características semelhantes pois um número maior de mensagens são repassadas para os automóveis. Além disso, observa-se que o protocolo *Spray and Wait* sobrecarrega

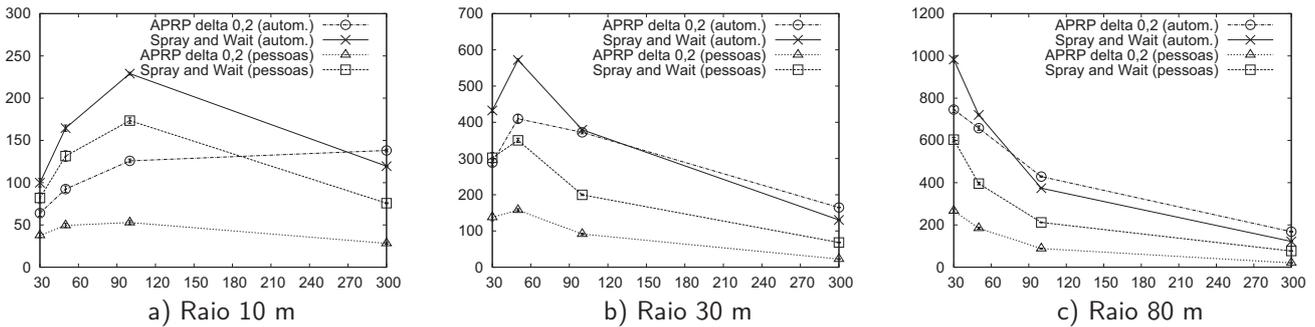


Figura 4.16: Mensagens recebidas por grupos de nodos com delta de 0,2. A Tabela 4.13 apresenta os valores numéricos dessa figura.

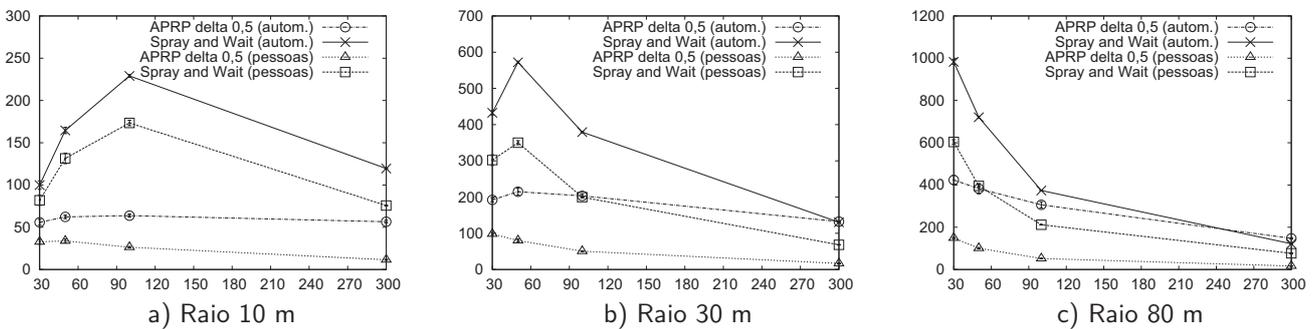


Figura 4.17: Mensagens recebidas por grupos de nodos com delta de 0,5. A Tabela 4.13 apresenta os valores numéricos dessa figura.

mais pessoas e automóveis nos cenários com poucos nodos se comparado ao APRP.

Ao considerar-se a razão das mensagens recebidas entre automóveis e pessoas para os dois protocolos na Figura 4.16, observa-se que a diferença é maior para o protocolo APRP. Isso indica que, no APRP com δ em 0,2, os nodos de maior potencial estão recebendo um percentual maior das cópias das mensagens comparativamente ao *Spray and Wait*. Contudo, apesar dessa sobrecarga, considerando o total de mensagens recebidas pelos dois protocolos, observa-se que o APRP gera um menor número de cópias das mensagens, conforme apresentado na Seção 4.4.3.

Quando se analisa os resultados da Figura 4.17, nota-se que o APRP com δ em 0,5 também sobrecarrega mais os automóveis do que as pessoas. No entanto, essa sobrecarga é menor do que o APRP com δ em 0,2 e do que o *Spray and Wait*, principalmente em redes esparsas com raio de alcance de 10 m. Dessa forma, tem-se uma opção de configuração do APRP caso haja necessidade de uma menor sobrecarga nos nodos de maior potencial.

4.4.7 Tabelas com Valores dos Gráficos

Percentual de Mensagens Entregues As Tabelas 4.4, 4.5 e 4.6 apresentam os valores numéricos para o percentual de mensagens entregues para os três padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente.

Tabela 4.4: Valores do percentual de mensagens entregues com *Random Waypoint*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	4,97	10,01	19,01	38,70
APRP delta 0,5	5,17	10,10	18,39	38,96
Epidêmico	5,22	10,72	22,47	31,33
PRoPHET	5,28	10,74	22,39	31,55
Spray and Wait	5,10	10,73	23,06	47,98
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	36,45	57,40	85,60	99,33
APRP delta 0,5	35,74	56,46	76,78	99,62
Epidêmico	37,26	53,92	53,72	37,75
PRoPHET	38,02	56,17	55,02	38,64
Spray and Wait	38,21	60,28	87,13	99,30
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	82,93	96,80	99,74	99,88
APRP delta 0,5	81,02	92,88	98,84	99,87
Epidêmico	61,10	59,03	50,05	35,60
PRoPHET	69,48	62,87	50,39	34,87
Spray and Wait	85,33	98,31	99,76	99,87

Tabela 4.5: Valores do percentual de mensagens entregues com *Shortest Path Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	45,62	64,56	87,71	99,01
APRP delta 0,5	44,36	63,62	80,92	90,71
Epidêmico	46,75	60,62	54,70	34,01
PRoPHET	46,72	63,06	62,81	36,94
Spray and Wait	47,45	67,65	89,41	98,93
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	87,03	98,24	99,81	99,88
APRP delta 0,5	84,08	94,27	98,97	99,62
Epidêmico	72,64	68,82	56,99	38,25
PRoPHET	78,69	74,00	58,02	36,13
Spray and Wait	88,12	99,53	99,86	99,91
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	98,96	99,76	99,89	99,94
APRP delta 0,5	95,78	99,16	99,74	99,83
Epidêmico	67,30	61,33	51,35	37,29
PRoPHET	73,27	63,82	50,65	32,86
Spray and Wait	99,82	99,85	99,90	99,92

Tabela 4.6: Valores do percentual de mensagens entregues com *Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	21,77	35,90	57,75	90,27
APRP delta 0,5	21,68	36,08	55,18	73,12
Epidêmico	20,49	33,09	49,50	47,00
PRoPHET	20,50	32,91	49,30	51,00
Spray and Wait	23,26	41,12	64,19	91,22
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	52,65	73,07	94,92	99,51
APRP delta 0,5	52,99	71,57	90,78	98,97
Epidêmico	49,88	59,44	57,41	43,41
PRoPHET	51,4	59,60	54,19	37,29
Spray and Wait	53,68	73,91	94,16	99,97
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	71,22	88,46	98,64	99,64
APRP delta 0,5	71,97	87,30	97,35	99,51
Epidêmico	50,90	51,67	47,86	38,77
PRoPHET	54,98	51,72	44,38	30,91
Spray and Wait	69,72	86,42	98,82	99,99

Overhead As Tabelas 4.7, 4.8 e 4.9 apresentam os valores numéricos para o *overhead* gerado na rede para os três padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente.

Tabela 4.7: *Overhead* gerado na rede com *Random Waypoint*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	0,32	0,38	1,09	4,75
APRP delta 0,5	0,32	0,27	0,43	1,26
Epidêmico	0,78	1,45	5,06	89,35
PRoPHET	0,73	1,38	4,76	72,83
Spray and Wait	0,75	1,35	3,13	8,65
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	0,46	1,29	3,45	4,75
APRP delta 0,5	0,24	0,39	0,89	2,32
Epidêmico	1,71	5,23	42,03	745,00
PRoPHET	1,54	4,67	33,70	515,56
Spray and Wait	1,50	2,91	5,35	5,00
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	2,34	3,48	4,40	4,77
APRP delta 0,5	0,65	1,06	1,87	3,36
Epidêmico	12,81	47,12	293,06	3626,93
PRoPHET	9,92	36,35	199,37	2329,26
Spray and Wait	4,19	4,79	4,87	4,96

Tabela 4.8: *Overhead* gerado na rede com *Shortest Path Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	0,45	0,80	2,23	4,49
APRP delta 0,5	0,06	0,09	0,31	1,12
Epidêmico	1,71	8,90	101,23	1749,48
PRoPHET	1,46	5,64	69,71	1191,36
Spray and Wait	1,35	2,78	5,14	5,01
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	1,25	2,57	4,20	4,75
APRP delta 0,5	0,19	0,40	1,17	2,49
Epidêmico	10,33	47,86	296,77	3521,15
PRoPHET	8,07	34,70	211,80	2425,20
Spray and Wait	3,34	4,63	4,87	4,96
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	2,80	3,77	4,42	4,73
APRP delta 0,5	0,54	0,93	1,81	2,94
Epidêmico	37,66	138,14	728,58	6112,80
PRoPHET	27,69	93,92	471,26	3878,23
Spray and Wait	4,51	4,74	4,87	4,95

Tabela 4.9: *Overhead* gerado na rede com *Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	0,41	1,00	2,14	4,75
APRP delta 0,5	0,22	0,33	0,63	1,79
Epidêmico	1,65	4,34	20,75	324,38
PRoPHET	1,37	3,45	14,32	173,54
Spray and Wait	1,35	2,59	5,27	5,42
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	1,41	2,93	4,12	4,79
APRP delta 0,5	0,64	1,05	1,79	3,51
Epidêmico	4,51	14,13	80,99	1111,94
PRoPHET	3,70	11,11	57,05	719,81
Spray and Wait	3,09	5,22	5,15	4,96
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	3,28	3,94	4,49	4,78
APRP delta 0,5	1,38	1,76	2,68	3,94
Epidêmico	22,27	68,61	329,21	3187,88
PRoPHET	15,37	47,61	222,07	2211,82
Spray and Wait	5,82	5,45	4,92	4,95

Atraso na Entrega das Mensagens As Tabelas 4.10, 4.11 e 4.12 apresentam os valores numéricos para o atraso na entrega das mensagens para os três padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement*, respectivamente.

Tabela 4.10: Atraso na entrega das mensagens com *Random Waypoint*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	19684,18	27350,26	42178,13	46941,84
APRP delta 0,5	19505,69	26972,63	43515,31	52188,50
Epidêmico	20597,38	28371,95	42569,84	33989,12
PRoPHET	21291,07	28617,97	42580,50	34585,82
Spray and Wait	20509,51	28460,54	43291,33	36123,43
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	18158,94	20534,52	15858,27	9012,12
APRP delta 0,5	18227,79	24344,68	27302,32	22249,33
Epidêmico	17771,47	17298,37	11661,58	6148,76
PRoPHET	17601,28	17329,44	11736,14	5929,26
Spray and Wait	17833,50	17133,20	9162,00	8722,28
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	6505,62	4957,73	3229,81	2363,87
APRP delta 0,5	10030,75	10309,86	8359,22	4703,43
Epidêmico	4950,94	3879,19	2860,00	1882,61
PRoPHET	5309,38	4151,71	2900,31	1823,04
Spray and Wait	4290,19	3054,21	2745,22	2511,35

Tabela 4.11: Atraso na entrega das mensagens com *Shortest Path Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	14581,46	17715,46	17710,66	9845,18
APRP delta 0,5	14860,04	19786,73	24226,63	23378,73
Epidêmico	13956,73	13364,30	7625,60	3185,96
PRoPHET	13951,96	14407,15	9418,26	3497,73
Spray and Wait	14000,61	14067,62	8474,89	7944,95
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	7123,87	4781,76	2516,44	1797,25
APRP delta 0,5	9371,37	10167,94	8115,54	4745,76
Epidêmico	4890,60	3554,25	2470,80	1637,77
PRoPHET	5319,41	3719,08	2412,64	1453,91
Spray and Wait	4424,73	2598,53	2003,49	1838,19
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	2716,32	1866,96	1373,74	1101,70
APRP delta 0,5	5825,82	5395,94	3560,51	2368,88
Epidêmico	2087,94	1678,23	1373,90	1153,58
PRoPHET	2173,51	1714,44	1342,02	932,70
Spray and Wait	1652,33	1384,07	1238,45	1261,74

Tabela 4.12: Atraso na entrega das mensagens com *Map Based Movement*.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	18049,33	23171,72	26840,71	21874,91
APRP delta 0,5	17852,52	25149,32	33215,34	36130,77
Epidêmico	13198,94	14281,44	14047,81	9054,93
PRoPHET	12967,04	14043,67	13861,17	9909,92
Spray and Wait	18905,78	22637,64	18625,91	17153,60
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	13739,56	12009,96	9427,79	6263,52
APRP delta 0,5	15617,60	18020,64	18581,18	11297,52
Epidêmico	10551,49	9608,99	7008,16	4215,80
PRoPHET	10084,49	8809,57	5873,23	2780,61
Spray and Wait	10927,17	7765,53	6593,73	7016,94
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2	7172,07	6775,37	5367,61	4288,51
APRP delta 0,5	11071,27	12135,87	10210,66	6329,44
Epidêmico	5589,82	4995,22	3977,58	2985,96
PRoPHET	5410,67	4418,60	3078,18	1631,47
Spray and Wait	4385,104	4371,57	5209,64	5231,30

Sobrecarga nos Nodos A Tabela 4.13 apresenta número médio de mensagens recebidas por grupos de nodos para o padrão de mobilidade *Map Based Movement*.

Tabela 4.13: Número médio de mensagens recebidas por grupos de nodos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2 (autom)	64,05	92,41	125,84	138,08
APRP delta 0,2 (pessoas)	37,69	49,47	52,86	28,23
APRP delta 0,5 (autom)	55,79	62,27	63,75	56,56
APRP delta 0,5 (pessoas)	32,78	33,82	26,38	11,50
Spray and Wait (autom)	99,92	164,54	229,01	119,50
Spray and Wait (pessoas)	81,90	131,46	173,42	75,68
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2 (autom)	288,68	409,78	372,21	164,81
APRP delta 0,2 (pessoas)	137,73	157,66	91,45	22,53
APRP delta 0,5 (autom)	192,39	214,82	203,39	132,28
APRP delta 0,5 (pessoas)	98,10	80,07	50,54	16,79
Spray and Wait (autom)	432,69	572,13	379,32	130,45
Spray and Wait (pessoas)	302,14	350,20	199,80	68,25
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP delta 0,2 (autom)	746,07	658,10	428,88	168,05
APRP delta 0,2 (pessoas)	266,53	184,61	88,40	20,26
APRP delta 0,5 (autom)	423,78	382,62	306,19	147,46
APRP delta 0,5 (pessoas)	149,23	100,70	52,03	16,64
Spray and Wait (autom)	982,67	720,60	373,76	122,02
Spray and Wait (pessoas)	603,61	395,79	212,00	76,64

4.5 Considerações Finais

O protocolo APRP, apresentado neste Capítulo, usa como principal métrica para decidir sobre o repasse das mensagens o potencial de entrega dos nodos, sendo que o potencial modifica seu valor à medida que muda a conectividade do nodo. Tal protocolo prioriza o repasse a nodos que tenham uma diferença de potencial, com relação ao seu, que justifique o repasse de uma mensagem. Além disso, ele realiza o controle do número de cópias das mensagens repassadas na rede.

Através de simulação foi mostrado que o protocolo APRP possui, na maioria dos casos, percentuais de mensagens entregues superiores aos dos protocolos Epidêmico e P_{Ro}PHET, mantendo níveis de entrega compatíveis com o protocolo *Spray and Wait*, utilizado como base do protocolo proposto.

Além de aumentar ou manter os níveis de entrega, o APRP gera uma considerável economia em relação a quantidade de cópias das mensagens. Essa economia ocorre em relação aos três protocolos utilizados nas simulações, Epidêmico, P_{Ro}PHET e *Spray and Wait*. Essa característica é muito importante para redes DTN, dada a limitação de recursos que tanto a rede quanto os nodos possuem. Além disso, mostrou-se experimentalmente que o protocolo APRP sobrecarrega o grupo de nodos que possui valores de potencial de entrega elevados, sendo que essa característica também ocorre de forma natural no protocolo *Spray and Wait*. A sobrecarga ocorre principalmente quando se utiliza um δ em 0,2, usado nos experimentos realizados por apresentar melhores níveis de atraso.

Dessa forma, o protocolo APRP é adequado para situações que se necessite economia de armazenamento e simplicidade de implementação, podendo suportar maior atraso na entrega das mensagens.

A seguir uma extensão ao protocolo APRP é proposta com o intuito de diminuir a sobrecarga nos nodos, principalmente de maior potencial, e melhorar o percentual de mensagens entregues.

5. O Protocolo APRP-ACK

O armazenamento em *buffers* das mensagens que precisam ser repassadas ocorre tanto em redes convencionais como em redes DTN. Quando os nodos de uma rede se tornam sobrecarregados e não possuem espaço em *buffer* para armazenar novas mensagens que chegam, políticas de descarte são usadas para liberar espaço para as novas mensagens. Um mecanismo utilizado para evitar o descarte é o envio de mensagens de *acknowledgment*. De acordo com [Lin09], o uso de *acknowledgments* permite que mensagens já entregues sejam removidas da rede, liberando recursos para outras mensagens, em um esforço de aumentar a probabilidade de entrega das mesmas. Nesse sentido, na tentativa de melhorar a eficiência do APRP, foi criado o APRP-Ack, o qual utiliza mensagens de *acknowledgment* para liberar espaço em *buffer* dos nodos. O objetivo é fazer com que principalmente os nodos mais sobrecarregados, de mais alto potencial, evitem o descarte de mensagens importantes, melhorando, assim, a probabilidade de entrega das mensagens e diminuindo a sobrecarga.

Nos protocolos de roteamento convencionais que utilizam tal mecanismo, cada vez que uma mensagem é entregue ao destino, uma mensagem de *acknowledgment* é emitida pela rede, sendo repassada por todos os nodos até a origem da mensagem entregue. Nesses protocolos, um caminho fim-a-fim existe e a mensagem de *acknowledgment* é transmitida através desse caminho. Já em redes DTN, como não há um caminho fim-a-fim entre origem e destino, uma estratégia diferente deve ser empregada. Uma forma seria a utilização do mecanismo de *flooding* para a transmissão dessa mensagem. Contudo, esse mecanismo acaba sobrecarregando os recursos da rede. Assim, para evitar essa sobrecarga, cada vez que um nodo entrega uma mensagem para seu destino final, o protocolo adiciona um *acknowledgment* na sua lista de mensagens. Dessa forma, os nodos não necessitam criar uma mensagem especial de *acknowledgment*. Quando dois nodos se encontram eles sincronizam suas listas de mensagens e verificam as mensagens confirmadas um do outro. O protocolo APRP-Ack utiliza esse mecanismo.

5.1 Algoritmo

A base do algoritmo do APRP-Ack continua sendo a mesma no APRP. Contudo, para o seu correto funcionamento é necessário a inclusão do *acknowledgment* para as mensagens entregues ao destino. O APRP-Ack inclui essa informação juntamente com a identificação e o número de créditos das mensagens mantidas pelo nodo, ou seja, essa informação é incluída no *summary vector* do nodo quando a mensagem for criada, no formato $\langle m, C, STATUS \rangle$. O *STATUS* indica se a mensagem foi entregue (“*ACK*”) ou se ainda está em *buffer* (“*BUF*”). Portanto, inicialmente as mensagens são configuradas com “*BUF*”.

O Algoritmo 5.1 apresenta o mecanismo de roteamento usado no APRP-Ack. Como sua base é igual ao protocolo APRP, estão destacadas (em negrito) no algoritmo as modificações necessárias para o seu correto funcionamento.

 Algoritmo 5.1: Mecanismo de roteamento do APRP-Ack.

```

procedure nodeMeetingNeighbor()
  obtain( $n, P_n, \text{summary\_vector\_}n, \text{time}$ )
  updateContactList( $\text{contact\_list\_}p, n, P_n, \text{time}, W_p$ )
  updateListAckedIds( $\text{summary\_vector\_}n, \text{summary\_vector\_}o$ )
  if ( $P_n > P_{maxO}$ ) then
     $P_{maxO} = P_n$ 
  end if
  calculate $P_o(\text{contact\_list\_}p, P_o)$ 
  if (nodeHasMessage( $\text{summary\_vector\_}o$ )) then
    directDelivery( $\text{summary\_vector\_}o, n, \text{summary\_vector\_}n$ )
    if ( $P_n \geq (P_o + \delta * (P_{maxO} - P_o))$ ) then
      sendMessageToNeighbor( $\text{summary\_vector\_}o, n, \text{summary\_vector\_}n$ )
    end if
  end if

procedure directDelivery( $\text{summary\_vector\_}o, n, \text{summary\_vector\_}n$ )
  for all  $m$  in  $\text{summary\_vector\_}o$  do
    if (( $\text{destination}(m) = n$ ) and (!neighborHasMessage( $m, \text{summary\_vector\_}n$ ))) then
      sendDirectMessage( $m, n$ )
      removeMessageAndUpdateList( $\text{summary\_vector\_}o, m, \text{"ACK"}$ )
    end if
  end for

procedure sendMessageToNeighbor( $\text{summary\_vector\_}o, n, \text{summary\_vector\_}n$ )

  for all  $\langle m, C \rangle$  in  $\text{summary\_vector\_}o$  do
    if (( $C > 1$ ) and (!neighborHasMessage( $m, \text{summary\_vector\_}n$ ))) then
      sendMessage( $n, m, \lfloor C/2 \rfloor$ )
       $C = \lfloor C/2 \rfloor$ 
      updateMsgList( $\text{summary\_vector\_}o, m, C$ )
    end if
  end for

procedure calculate $P_o(\text{contact\_list\_}p, P_o)$  {Calcula o potencial do nodo origem.}
procedure createNewMessage( $m, \text{summary\_vector\_}o, C$ ) {Inclui a identificação da nova mensagem na lista de mensagens, configura o número de créditos da mensagem, bem como seu STATUS.}
procedure neighborHasMessage( $m, \text{summary\_vector\_}n$ ) {Verifica se a mensagem  $m$  está presente no  $\text{summary vector}$  do nodo encontrado. Caso esteja retorna true, caso contrário retorna false.}
procedure nodeHasMessage( $\text{summary\_vector\_}o$ ) {Verifica se há mensagens para serem entregues no  $\text{summary vector}$  no próprio nodo, retornando true ou false.}
procedure obtain( $n, P_n, \text{summary\_vector\_}n, \text{time}$ ) {Retorna a identificação, o potencial e o  $\text{summary vector}$  do nodo encontrado, bem como o tempo do encontro.}
procedure removeMessageAndUpdateList( $\text{summary\_vector\_}o, m, \text{"ACK"}$ ) {Remove do buffer a mensagem entregue ao destino e altera o STATUS dessa mensagem para "ACK" no  $\text{summary vector}$  do próprio nodo.}
procedure sendMessage( $n, m, C$ ) {Repassa mensagem para o nodo.}
procedure sendDirectMessage( $m, n$ ) {Envia mensagem para o destino.}
procedure updateContactList( $\text{contact\_list\_}p, n, P_n, \text{time}, W_p$ ) {Inclui a identificação do nodo encontrado na lista de contatos e remove contatos antigos de acordo com a janela de tempo.}
procedure updateListAckedIds( $\text{summary\_vector\_}n, \text{summary\_vector\_}o$ ) {Atualiza o  $\text{summary vector}$  do próprio nodo com as mensagens confirmadas que estão no  $\text{summary vector}$  do nodo encontrado, além de remover do buffer as mensagens confirmadas que foram atualizadas.}
procedure updateMsgList( $\text{summary\_vector\_}o, m, C$ ) {Atualiza a lista de mensagens dando o novo crédito para a mensagem repassada.}
  
```

Nesse algoritmo foram incluídos dois procedimentos: $updateListAckedIds(summary_vector_n, summary_vector_o)$ e $removeMessageAndUpdateList(summary_vector_o, m, "ACK")$. O $updateListAckedIds(summary_vector_n, summary_vector_o)$ é responsável por fazer a sincronização e remoção das mensagens com o *STATUS* igual a “ACK” do *buffer* do nodo. Já o $removeMessageAndUpdateList(summary_vector_o, m, "ACK")$ é responsável por fazer a remoção da mensagem entregue ao destino final, bem como a alteração de seu *STATUS* no *summary vector* do nodo.

5.2 Avaliação Experimental

A avaliação do APRP-Ack também foi realizada através do simulador ONE, sendo utilizados os mesmos parâmetros descritos na Seção 4.4. Os experimentos foram conduzidos com 5000 e 10000 mensagens no mesmo tempo de simulação. Realizou-se experimentos com 10000 mensagens para verificar o desempenho dos protocolos com uma carga maior de mensagens.

Devido aos resultados comentados na Seção 4.4 sobre o desempenho do APRP, em comparação com os protocolos Epidêmico, P_{Ro}PHET e *Spray and Wait*, observou-se que ambos APRP e *Spray and Wait* apresentam um melhor percentual de mensagens entregues, bem como *overhead* mais baixo. Portanto, nesta Seção o APRP-Ack será comparado apenas a esses dois protocolos.

Além disso, apesar de experimentos com os padrões de mobilidade *Random Waypoint*, *Shortest Path Map Based Movement* e *Map Based Movement* serem realizados, apenas resultados com o padrão de mobilidade *Map Based Movement* serão apresentados, pois os outros não trazem informações adicionais para o entendimento do comportamento do protocolo proposto.

A seguir estão apresentados os resultados com o conjunto básico de métricas usado para avaliar o desempenho dos protocolos de roteamento para redes DTN: percentual de mensagens entregues, *overhead* e atraso médio na entrega das mensagens. Por outro lado, estão apresentados os resultados com o conjunto adicional de métricas apresentado na Seção 2.5: ocupação de *buffer*, *buffer overflow*, entregas redundantes e tempo de vida das mensagens. Esse conjunto de métricas adicional foi utilizado para compreender o impacto da estratégia de *acknowledgment* no uso dos *buffers*, bem como entender melhor o comportamento do conjunto básico de métricas.

5.2.1 Percentual de Mensagens Entregues

As Figuras 5.1 e 5.2 apresentam o percentual de mensagens entregues para os experimentos realizados com 5000 e 10000 mensagens, respectivamente. Observa-se que o percentual de mensagens entregues é semelhante em todos os protocolos. A diferença é mais visível quando é realizada a simulação com 10000 mensagens, como pode ser visto na Figura 5.2b e c. Na Figura 5.2c, o APRP-Ack supera em média os protocolos APRP e *Spray and Wait* em 4,49% e 6,07%, respectivamente. Como mais mensagens estão na rede, usando recursos dos nodos, os *acknowledgments* ajudarão a liberar espaço em *buffer*, evitarão que mensagens sejam descartadas por *buffer overflow*,

e aumentarão o percentual de mensagens entregues. Isso pode ser melhor observado na análise realizada com as métricas adicionais.

Uma característica comum que ocorre nos três protocolos analisados, independente do número de mensagens enviadas, é o aumento no percentual de mensagens entregues à medida que a rede se torna mais densa, seja por um maior número de nodos ou devido a um maior raio de comunicação.

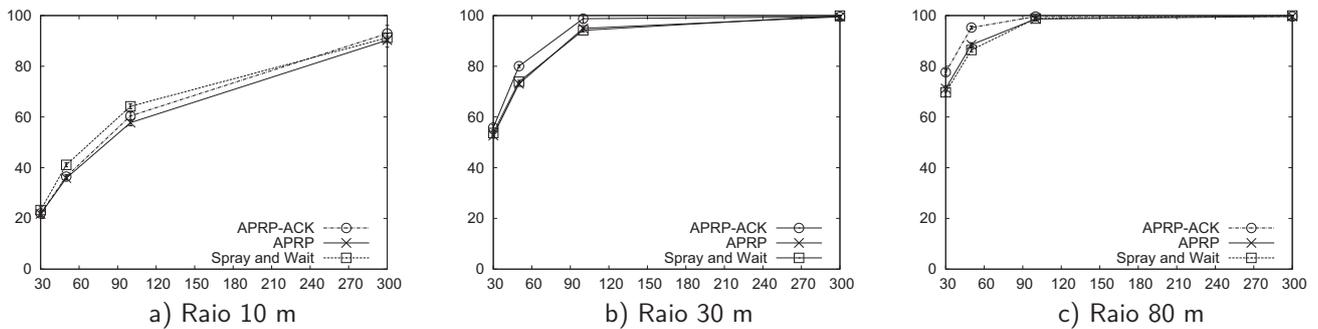


Figura 5.1: Percentual de mensagens entregues para 5000 mensagens. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 5.1 apresenta os valores numéricos dessa figura.

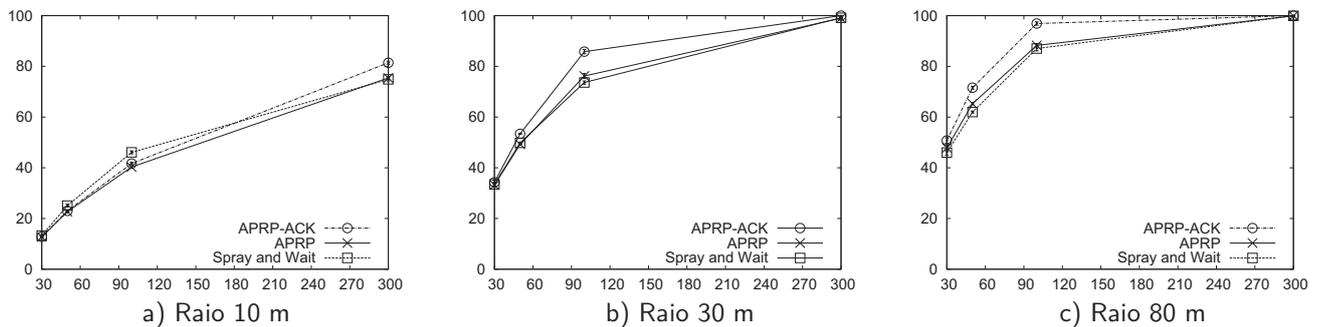


Figura 5.2: Percentual de mensagens entregues para 10000 mensagens. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 5.2 apresenta os valores numéricos dessa figura.

5.2.2 Overhead

As Figuras 5.3 e 5.4 apresentam o *overhead* para 5000 e 10000 mensagens, respectivamente, sendo que o *overhead* do APRP-Ack é menor do que nos demais.

Observa-se que, com a utilização de *acknowledgments*, o *overhead* que já era baixo no APRP se torna mais baixo ainda no APRP-Ack. Isso ocorre pois aquelas mensagens que já foram confirmadas não são repassadas para outros nodos, reduzindo o número de cópias das mensagens na rede. Como um exemplo, para os resultados apresentados na Figura 5.4b, o APRP-Ack reduziu o *overhead* em média 8,43% e 42,56% em comparação ao APRP e *Spray and Wait*, respectivamente.

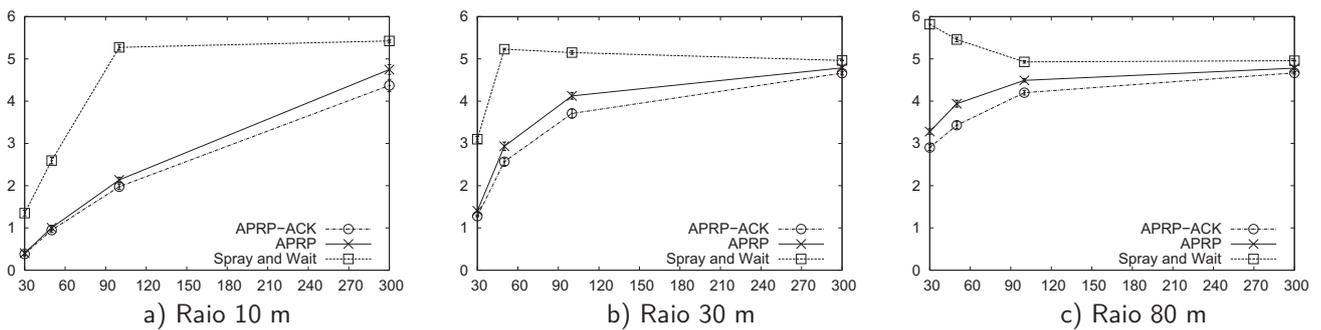


Figura 5.3: *Overhead* gerado na rede para 5000 mensagens. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 5.3 apresenta os valores numéricos dessa figura.

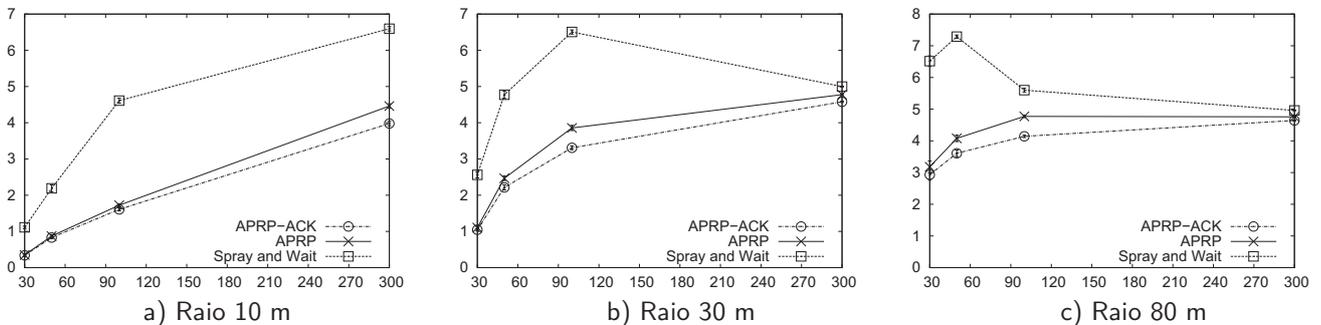


Figura 5.4: *Overhead* gerado na rede para 10000 mensagens. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 5.4 apresenta os valores numéricos dessa figura.

5.2.3 Atraso na Entrega das Mensagens

Os valores obtidos em relação ao atraso médio na entrega das mensagens estão apresentados nas Figuras 5.5 e 5.6 para 5000 e 10000 mensagens, respectivamente.

Observa-se que em geral o APRP-Ack apresenta um maior atraso. Como um exemplo, para os resultados apresentados na Figura 5.6b, o APRP-Ack gerou um aumento no *atraso* em média de 8,94% e 45,67%, em comparação aos protocolos APRP e *Spray and Wait*, respectivamente. Uma razão provável é o menor número de cópias (*overhead*) e o mecanismo de *acknowledgment*, uma vez que os *buffers* são menos usados e as mensagens podem ser armazenadas por mais tempo - até que o destino seja alcançado. Isto está de acordo com o fato do atraso ser computado somente sobre mensagens entregues. Além disso, o repasse ocorre apenas quando o nodo encontrado possui um potencial de entrega que justifique o repasse, não ocorrendo para qualquer nodo encontrado.

Quando a rede é muito esparsa, com raio de alcance de 10 m e poucos nodos, o atraso dos três protocolos é praticamente o mesmo, como ocorre na Seção 4.4.4. Já em redes mais densas, com 300 nodos e raio de alcance de 30 m e 80 m, o atraso se torna menor que o *Spray and Wait* e equivalente ao APRP.

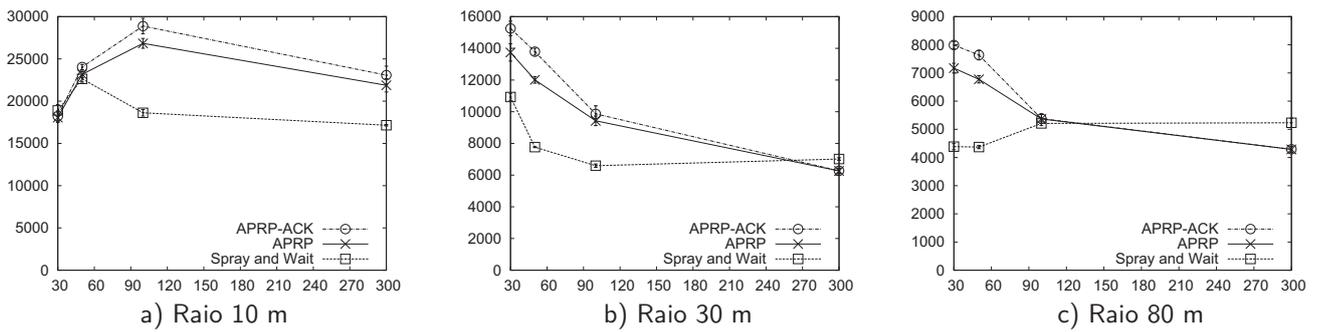


Figura 5.5: Atraso na entrega das mensagens para 5000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.5 apresenta os valores numéricos dessa figura.

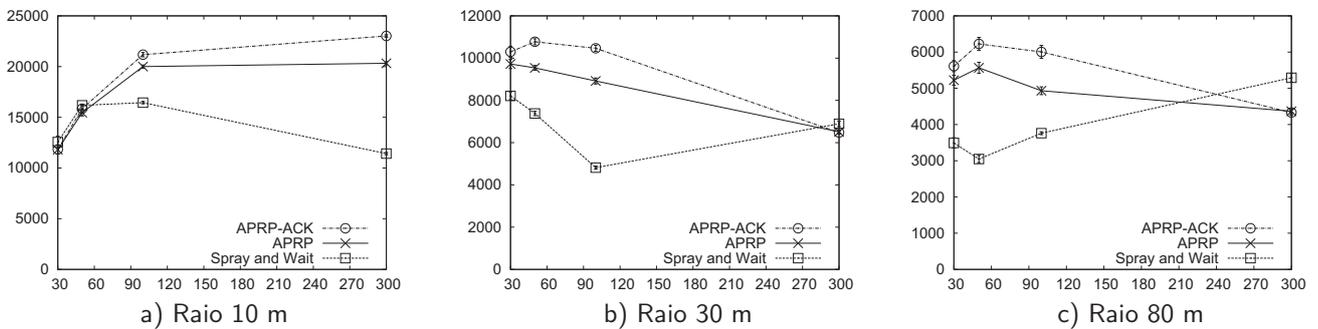


Figura 5.6: Atraso na entrega das mensagens para 10000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.6 apresenta os valores numéricos dessa figura.

5.2.4 Ocupação de Buffer

As Figuras 5.7 e 5.8 apresentam a ocupação média dos *buffers* dos nodos da rede para 5000 e 10000 mensagens, respectivamente.

Observa-se que, geralmente, os nodos que usam APRP-Ack tem um número de mensagens no *buffer* muito semelhante ao APRP. Contudo, quando se analisa o percentual de mensagens entregues, verifica-se que o APRP-Ack entrega mais do que o APRP. Isso acontece porque o mecanismo de *acknowledgment* ajuda a liberar os *buffers* e permite que mensagens ainda não entregues o utilizem.

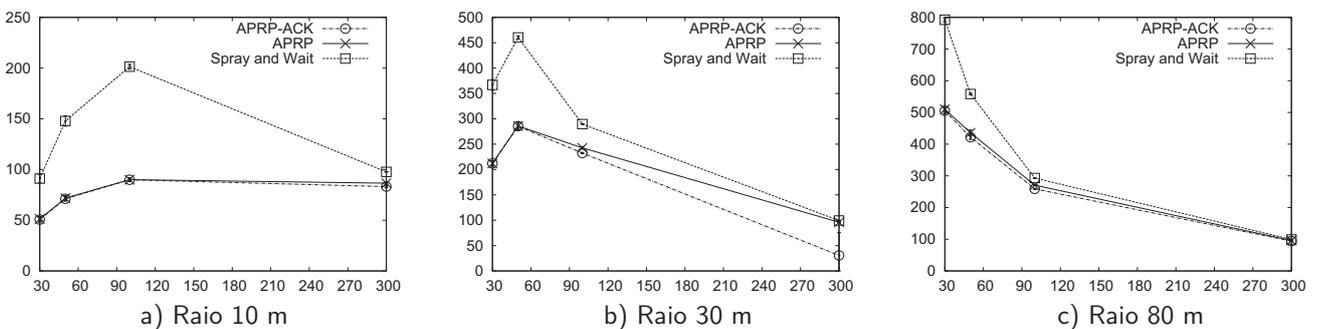


Figura 5.7: Ocupação de *buffer* para 5000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.7 apresenta os valores numéricos dessa figura.

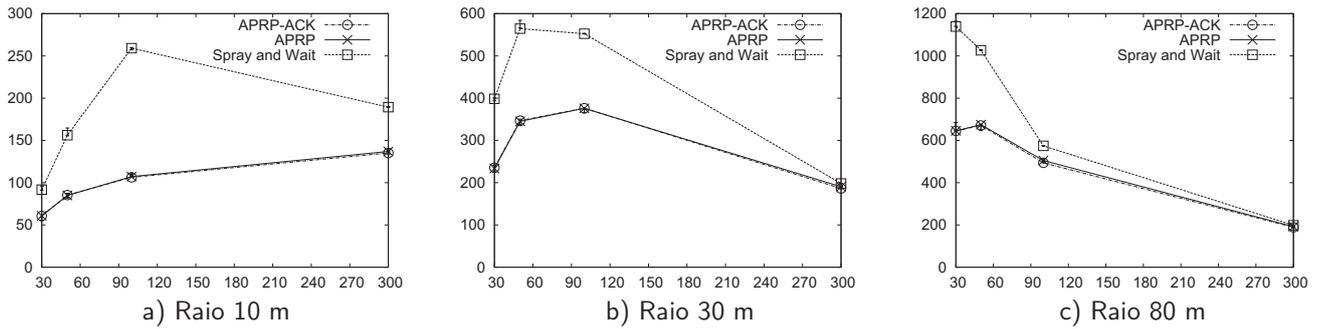


Figura 5.8: Ocupação de *buffer* para 10000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.8 apresenta os valores numéricos dessa figura.

5.2.5 Mensagens Descartadas Devido a Buffer Overflow

As Figuras 5.9 e 5.10 apresentam a quantidade de mensagens descartadas devido a *buffer* cheio, tanto para 5000 quanto para 10000 mensagens, respectivamente.

O *overflow* gerado pelo APRP-Ack é menor que o gerado pelo APRP e pelo *Spray and Wait*. Essa característica ocorre pela utilização de mensagens de *acknowledgment*, que libera espaço das mensagens já entregues e ajuda a não sobrecarga dos *buffers*.

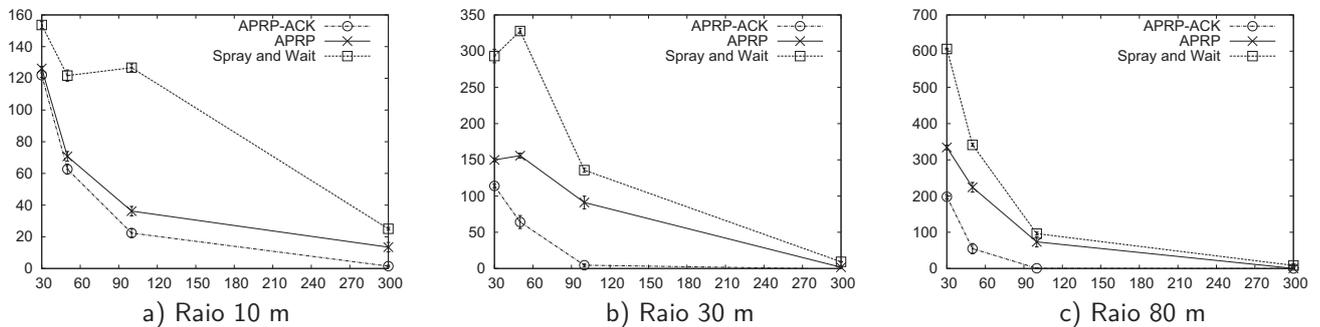


Figura 5.9: Mensagens descartadas devido a *buffer overflow* para 5000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.9 apresenta os valores numéricos dessa figura.

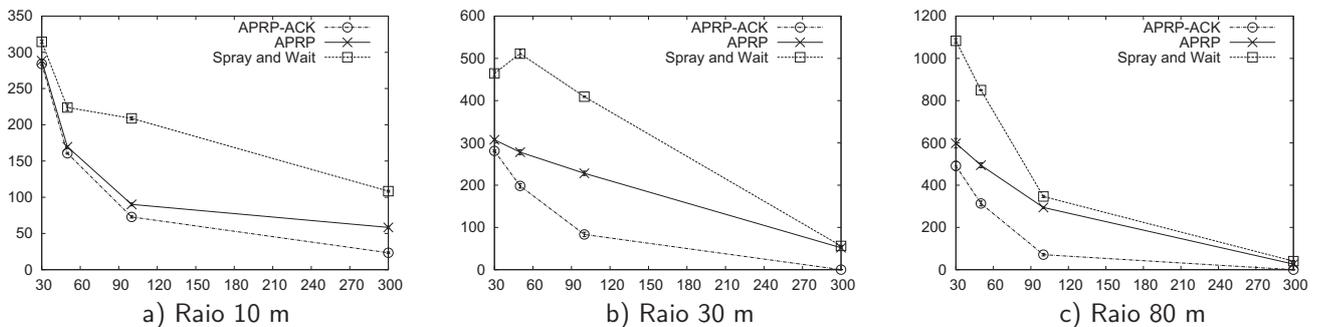


Figura 5.10: Mensagens descartadas devido a *buffer overflow* para 10000 mensagens. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.10 apresenta os valores numéricos dessa figura.

5.2.6 Entregas Redundantes

De acordo com o que foi comentado na Seção 2.5, essa é uma métrica que mede a redundância na entrega das mensagens. As Figuras 5.11 e 5.12 apresentam essa redundância para 5000 e 10000 mensagens, respectivamente. Pode-se visualizar através das figuras que o APRP-Ack repassa menos mensagens depois da primeira entrega quando comparado com os protocolos APRP e o *Spray and Wait*. Como exemplo, pode-se observar na Figura 5.12b que em média 10% das entregas do APRP-Ack são redundantes, quando a rede tem 100 nodos, enquanto o *Spray and Wait* tem 20% e o APRP tem 30%.

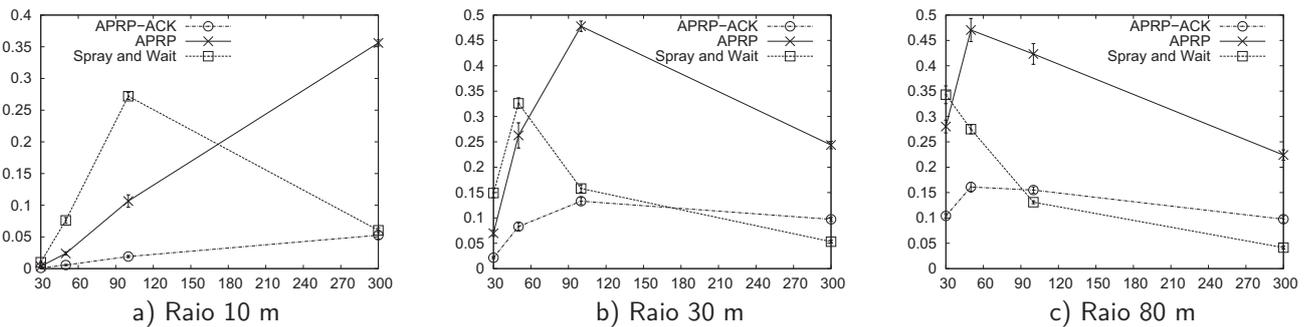


Figura 5.11: Entregas redundantes para 5000 mensagens. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.11 apresenta os valores numéricos dessa figura.

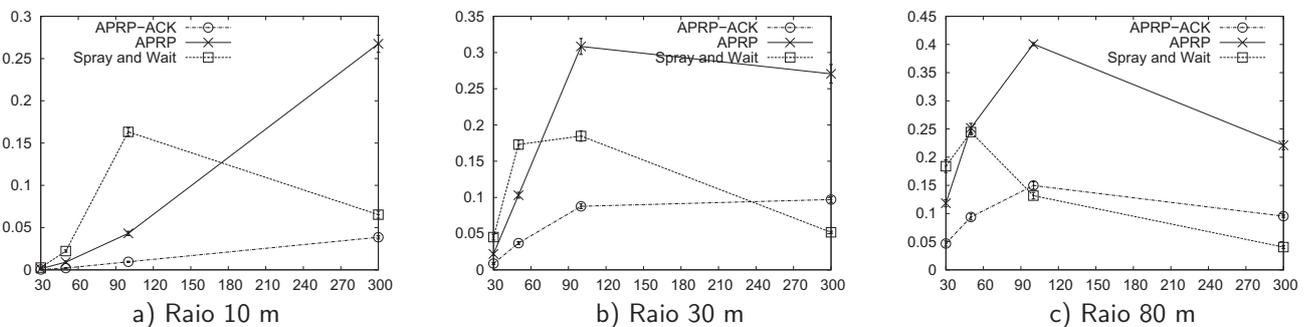


Figura 5.12: Entregas redundantes para 10000 mensagens. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 5.12 apresenta os valores numéricos dessa figura.

5.2.7 Tempo de Vida das Mensagens

As Figuras 5.13 e 5.14 apresentam o tempo de vida para 5000 e 10000 mensagens, respectivamente. Os resultados mostram que o tempo de vida das mensagens com o protocolo APRP-Ack é consideravelmente menor que o tempo registrado pelos protocolos APRP e *Spray and Wait*. Pode-se observar uma maior diferença quando a densidade da rede e o tráfego são maiores, como 100 e 300 nodos e para raios de alcance de 30 m e 80 m. Isso é um impacto direto das mensagens de *acknowledgment* usadas pelo protocolo APRP-Ack e mostram o seu benefício.

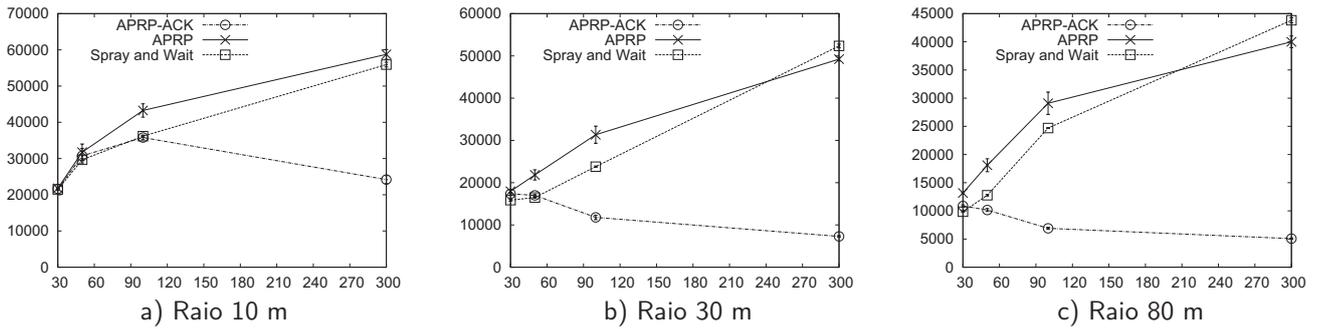


Figura 5.13: Tempo de vida das mensagens para 5000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.13 apresenta os valores numéricos dessa figura.

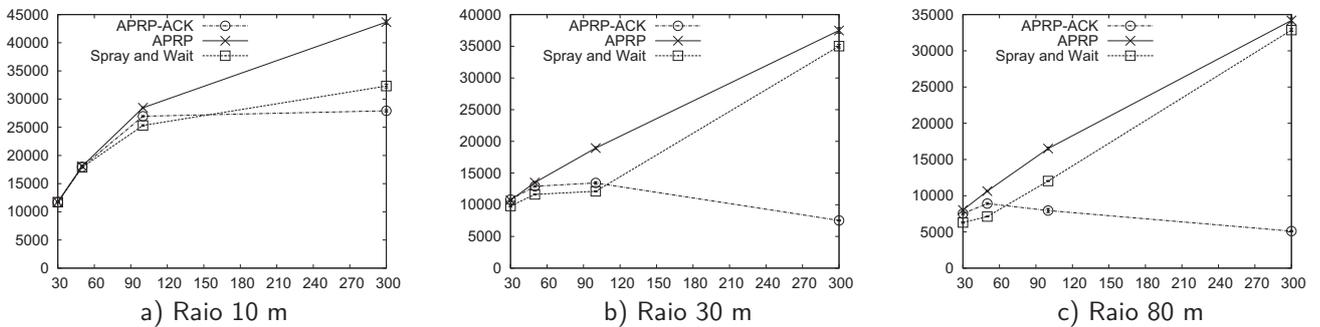


Figura 5.14: Tempo de vida das mensagens para 10000 mensagens. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 5.14 apresenta os valores numéricos dessa figura.

5.2.8 Tabelas com Valores dos Gráficos

Percentual de Mensagens Entregues As Tabelas 5.1 e 5.2 apresentam os valores numéricos para o percentual de mensagens entregues para 5000 e 1000 mensagens, respectivamente.

Tabela 5.1: Valores do percentual de mensagens entregues para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	21,95	36,48	60,43	92,93
APRP	21,77	35,90	57,75	90,27
Spray and Wait	23,26	41,12	64,19	91,22
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	55,83	80,06	98,77	99,51
APRP	52,65	73,07	94,92	99,51
Spray and Wait	53,68	73,91	94,16	99,97
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	77,67	95,26	99,60	99,64
APRP	71,22	88,46	98,64	99,64
Spray and Wait	69,72	86,42	98,82	99,99

Tabela 5.2: Valores do percentual de mensagens entregues para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	12,77	23,02	41,67	81,41
APRP	12,74	22,68	40,30	75,43
Spray and Wait	13,29	25,10	46,09	74,87
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	34,27	53,41	85,80	99,99
APRP	33,18	49,24	76,18	99,00
Spray and Wait	33,47	49,83	73,63	99,16
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	50,79	71,58	96,89	100,00
APRP	47,85	65,25	88,25	99,93
Spray and Wait	46,02	61,98	87,04	99,92

Overhead As Tabelas 5.3 e 5.4 apresentam os valores numéricos para o *overhead* gerado na rede para 5000 e 1000 mensagens, respectivamente.

Tabela 5.3: *Overhead* gerado na rede para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,38	0,95	1,98	4,37
APRP	0,41	1,00	2,14	4,75
Spray and Wait	1,35	2,59	5,27	5,42
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	1,28	2,57	3,71	4,66
APRP	1,41	2,93	4,12	4,79
Spray and Wait	3,09	5,22	5,15	4,96
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	2,90	3,43	4,20	4,67
APRP	3,28	3,94	4,49	4,78
Spray and Wait	5,82	5,45	4,92	4,95

Tabela 5.4: *Overhead* gerado na rede para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,34	0,83	1,61	3,98
APRP	0,35	0,87	1,72	4,46
Spray and Wait	1,11	2,19	4,61	6,59
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	1,05	2,21	3,31	4,58
APRP	1,10	2,47	3,86	4,78
Spray and Wait	2,56	4,77	6,51	5,00
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	2,93	3,61	4,14	4,64
APRP	3,18	4,08	4,77	4,76
Spray and Wait	6,51	7,29	5,60	4,96

Atraso na Entrega das Mensagens As Tabelas 5.5 e 5.6 apresentam os valores numéricos para o atraso na entrega das mensagens para 5000 e 1000 mensagens, respectivamente.

Tabela 5.5: Atraso na entrega das mensagens para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	18224,24	24035,80	28882,78	23076,55
APRP	18049,33	23171,72	26840,71	21874,91
Spray and Wait	18905,78	22637,64	18625,91	17153,60
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	15251,17	13787,35	9864,25	6273,39
APRP	13739,56	12009,96	9427,79	6263,52
Spray and Wait	10927,17	7765,53	6593,73	7016,94
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	7991,94	7637,38	5380,02	4292,35
APRP	7172,07	6775,37	5367,61	4288,51
Spray and Wait	4385,104	4371,57	5209,64	5231,30

Tabela 5.6: Atraso na entrega das mensagens para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	11832,33	15848,47	21173,57	23016,37
APRP	11805,87	15454,31	20001,24	20324,07
Spray and Wait	12557,23	16178,12	16444,06	11421,08
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	10298,90	10773,79	10467,38	6479,34
APRP	9722,25	9536,42	8919,06	6511,92
Spray and Wait	8213,47	7382,18	4816,54	6890,32
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	5612,24	6225,86	6004,91	4332,40
APRP	5223,46	5566,07	4932,66	4367,27
Spray and Wait	3492,41	3047,96	3761,48	5291,62

Ocupação de Buffer As Tabelas 5.7 e 5.8 apresentam os valores numéricos para a ocupação de *buffer* para 5000 e 1000 mensagens, respectivamente.

Tabela 5.7: Ocupação de *buffer* para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	50,64	71,12	89,87	83,12
APRP	51,25	71,80	90,00	86,50
Spray and Wait	91,06	147,79	201,33	97,66
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	212,01	285,50	232,43	30,49
APRP	211,95	285,71	242,65	96,01
Spray and Wait	366,81	460,36	289,55	99,36
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	505,21	421,95	258,94	94,14
APRP	509,76	436,08	270,53	95,99
Spray and Wait	792,24	558,08	292,95	99,29

Tabela 5.8: Ocupação de *buffer* para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	60,67	85,32	106,35	135,11
APRP	60,80	84,84	107,22	136,90
Spray and Wait	91,77	156,14	259,08	189,31
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	235,47	346,90	376,28	186,28
APRP	233,60	345,12	375,80	190,41
Spray and Wait	398,40	564,50	552,52	198,18
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	645,03	670,64	493,90	188,66
APRP	645,17	674,46	505,01	192,16
Spray and Wait	1138,83	1026,38	574,11	198,47

Mensagens Descartadas devido a Buffer Overflow As Tabelas 5.9 e 5.10 apresentam os valores numéricos para as mensagens descartadas devido a *buffer overflow* para 5000 e 1000 mensagens, respectivamente.

Tabela 5.9: Mensagens descartadas devido a *buffer overflow* para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	122,01	62,70	22,39	1,49
APRP	126,22	70,76	36,22	13,48
Spray and Wait	153,67	121,73	126,72	25,06
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	113,97	64,13	4,50	0,67
APRP	149,92	155,80	91,10	1,70
Spray and Wait	293,30	327,78	135,78	9,28
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	198,05	54,74	0,70	0,00
APRP	334,14	224,26	73,39	0,29
Spray and Wait	606,02	341,25	96,28	8,62

Tabela 5.10: Mensagens descartadas devido a *buffer overflow* para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	284,09	160,68	72,84	23,33
APRP	288,39	169,65	90,10	58,35
Spray and Wait	314,51	223,89	208,82	108,36
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	281,04	198,30	83,34	0,15
APRP	307,56	278,09	228,11	51,96
Spray and Wait	464,42	511,28	409,35	56,18
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	491,24	313,22	71,02	0,01
APRP	598,19	494,45	294,27	26,70
Spray and Wait	1083,10	849,84	346,46	40,56

Entregas Redundantes As Tabelas 5.11 e 5.12 apresentam os valores numéricos para as mensagens que foram entregues de forma redundante para 5000 e 1000 mensagens, respectivamente.

Tabela 5.11: Entregas redundantes para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,00	0,01	0,02	0,05
APRP	0,00	0,02	0,11	0,36
Spray and Wait	0,01	0,08	0,27	0,06
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,02	0,08	0,13	0,10
APRP	0,07	0,26	0,48	0,24
Spray and Wait	0,15	0,33	0,16	0,05
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,10	0,16	0,16	0,10
APRP	0,28	0,47	0,42	0,22
Spray and Wait	0,34	0,28	0,13	0,04

Tabela 5.12: Entregas redundantes para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,00	0,00	0,01	0,04
APRP	0,00	0,01	0,04	0,27
Spray and Wait	0,00	0,02	0,16	0,07
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,01	0,04	0,09	0,10
APRP	0,02	0,10	0,31	0,27
Spray and Wait	0,05	0,17	0,18	0,05
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	0,05	0,09	0,15	0,10
APRP	0,12	0,25	0,40	0,22
Spray and Wait	0,18	0,24	0,13	0,04

Tempo de Vida das Mensagens As Tabelas 5.13 e 5.14 apresentam os valores do tempo de vida das mensagens para 5000 e 1000 mensagens, respectivamente.

Tabela 5.13: Tempo de vida das mensagens para 5000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	21614,36	30746,04	35788,81	24178,70
APRP	21779,90	31742,67	43266,19	58757,89
Spray and Wait	21470,12	29652,47	36180,19	55932,97
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	17370,73	16996,67	11785,11	7292,27
APRP	17954,09	21838,23	31349,42	49224,14
Spray and Wait	15861,42	16491,45	23783,33	52357,51
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	10865,00	10145,74	6902,89	5078,32
APRP	13167,17	18092,73	29095,96	39996,45
Spray and Wait	9852,22	12762,45	24696,79	43822,21

Tabela 5.14: Tempo de vida das mensagens para 10000 mensagens.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	11710,21	17964,36	26957,68	27923,70
APRP	11745,83	18098,99	28466,25	43658,84
Spray and Wait	11707,98	17878,11	25318,29	32316,18
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	10824,16	12914,40	13439,71	7521,80
APRP	10716,47	13511,26	18941,30	37487,10
Spray and Wait	9801,75	11633,69	12123,70	35015,30
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Ack	7536,18	8918,09	7951,23	5112,67
APRP	8035,64	10636,05	16511,95	34193,50
Spray and Wait	6300,53	7127,85	12026,24	32865,12

5.3 Considerações Finais

Neste Capítulo o protocolo APRP-Ack foi apresentado. Tal protocolo é uma extensão do protocolo APRP e usa mensagens de *acknowledgment* com o intuito de liberar espaço em *buffer* de mensagens já entregues, principalmente dos nodos de alto potencial, para aumentar os níveis de entrega.

De acordo com resultados de simulação, observa-se que o APRP-Ack, na maioria das configurações, possui um maior percentual de mensagens entregues se comparado aos protocolos APRP e *Spray and Wait*, e gera uma menor carga de cópias das mensagens na rede. Dessa forma, ele se mostra com desempenho superior ao APRP e *Spray and Wait* nesses quesitos, principalmente quando um maior número de mensagens está presente na rede.

Contudo, o atraso médio na entrega das mensagens é mais elevado, já que o repasse não acontece imediatamente no próximo encontro com outro nodo, mas depende também do nodo encontrado ter um potencial de entrega que justifique o repasse. Além disso, o mecanismo de *acknowledgment* faz com que as mensagens não sejam descartadas por *buffer overflow*, permanecendo mais tempo em *buffer* até que possam ser entregues.

A seguir é definida uma variação para o APRP-Ack, com o intuito de reduzir o atraso médio na entrega das mensagens, sem prejudicar o percentual de mensagens entregues.

6. O Protocolo APRP-Group

Dada a avaliação dos protocolos APRP e APRP-Ack, em especial o elevado atraso gerado por esses protocolos (conforme Capítulos 4 e 5), foi criado o APRP-Group. O objetivo deste protocolo é diminuir o atraso médio na entrega, sem prejudicar a taxa de entrega das mensagens. Para tanto, o APRP-Group se utiliza de uma característica observada a partir dos potenciais de entrega dos nodos. Observou-se que os valores de potenciais de entrega tendem a estar agrupados, dependendo do padrão de movimentação dos nodos. Dessa forma, o APRP-Group agrupa esses valores, sendo que o repasse das mensagens ocorre considerando os potenciais dos grupos, ao invés de considerar o potencial individual do nodo. Além disso, com o objetivo de aumentar a disseminação de mensagens na rede, foi definido um novo mecanismo para administração de cópias de uma mensagem.

6.1 Caracterização e Agrupamento de Potenciais

O conceito de grupos de potenciais é baseado na observação de que, em muitas situações, os nodos da rede podem ser agrupados de acordo com o seu potencial de entrega. Conforme discutido na Seção 4.1, observou-se que grupos de nodos com padrão de mobilidade semelhante tendem a valores de potenciais médios aproximados. Os nodos que têm maior mobilidade possuem, em geral, um valor de potencial mais elevado, pois seu contato com outros nodos é maior. Já os nodos que se movimentam menos possuem esse valor reduzido. A distribuição de potenciais para dois tipos de nodos pode ser visualizada na Figura 6.1.

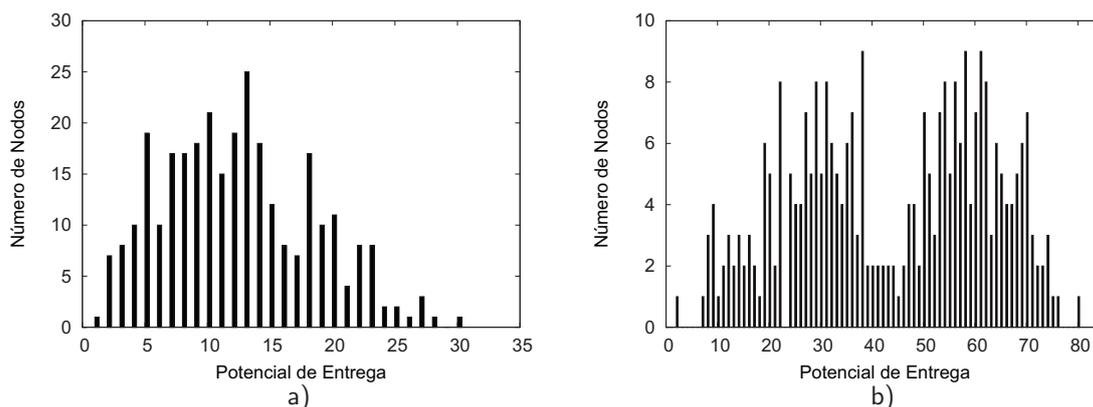


Figura 6.1: Distribuição de potenciais para dois tipos de nodos. a) Resultado da simulação com intervalo de velocidades aproximado. b) Resultado da simulação com maior diferença no intervalo de velocidades.

A Figura 6.1a apresenta os resultados de potencial para dois tipos de nodos quando seus intervalos de velocidade são aproximados. Nesse caso, um tipo varia a velocidade entre $[0.5, 1.0]$ m/s, enquanto o outro varia entre $[2.0, 5.0]$ m/s. Já a Figura 6.1b apresenta os potenciais quando há uma maior diferença no intervalo das velocidades, com intervalos entre $[0.5, 1.0]$ m/s e $[20.0, 22.0]$ m/s.

Observa-se que quanto maior é a diferença entre os intervalos de velocidades dos tipos de nodos, melhor é o reconhecimento de grupos.

A identificação de grupos é um processo distribuído, que deve rodar em cada nodo da rede. A cada encontro, os nodos trocam seus valores de potenciais, colecionando amostras de <nodo, potencial>. Com o tempo, um nodo coleciona valores de potenciais de outros nodos (chamada de lista de contatos) e com, essa coleção, os grupos podem ser identificados. A identificação de grupos pode utilizar técnicas estatísticas, reconhecimento de padrões e mineração de dados [Ber02]. Muitas dessas técnicas são utilizadas para se agrupar uma quantidade bastante grande de dados, como milhões de dados [Xu05]. Esse não é o caso dos dados de potenciais coletados pelos nodos, então o protocolo proposto utiliza uma abordagem com programação dinâmica, adaptada de [Oli08] e [Ski97]. Tal abordagem minimiza tanto os desvios padrão de cada grupo como a diferença entre esses desvios. Para tanto, são analisadas as divisões em todos os pontos de corte possíveis para k grupos, como descrito abaixo.

Assume-se um conjunto de valores de potenciais e um valor inteiro k de número de grupos a serem formados. Para otimizar o processo, os valores dos desvios calculados vão sendo armazenados em uma matriz M , de forma que esses valores não precisem ser calculados novamente. Cada elemento $M_{n,k}$ representa o menor valor de um conjunto. Cada elemento desse conjunto é o maior desvio padrão de um grupo para uma dada divisão de n valores em k grupos. Esse conjunto cobre todos os possíveis divisores de n valores em k grupos.

O algoritmo utilizado para essa função está descrito a seguir (Algoritmo 6.1). Tal algoritmo utiliza as seguintes estruturas e variáveis:

- *frequency_list*: armazena a distribuição de frequência dos potenciais que estão na lista de contato do nodo;
- k : número de grupos a serem divididos;
- M : matriz auxiliar que armazena os desvios padrão para um determinado valor de k ;
- *break_list*: matriz que armazena as posições de corte para um determinado k , bem como para valores de k inferiores.

Para que ocorra o processo de divisão em grupos, deve-se calcular a distribuição de frequência dos valores de potencial armazenados na lista de contato do nodo. Tal distribuição de frequência (*frequency_list*) é passada como parâmetro para o procedimento *cluster(frequency_list, k, break_list)*, assim como o número k de grupos que deseja-se dividir. O procedimento retorna a matriz *break_list* com as posições ótimas de corte para k grupos.

Inicialmente, tal procedimento inicializa a primeira linha e coluna da matriz M . A primeira coluna recebe os valores dos desvios padrão supondo-se que haja somente um grupo, ou seja, $k = 1$. O procedimento *deviation(begin, end, frequency_list)*, é utilizado para calcular o desvio padrão para os valores de potencial que estão na *frequency_list*, a partir de uma posição inicial (*begin*)

Algoritmo 6.1: Algoritmo utilizado no agrupamento dos potenciais, baseado em [Ski97].

```

procedure cluster(frequency_list, k, break_list)
  for i = 1 to n do
    M[i][1] = deviation(1, i, frequency_list)
  end for
  for i = 1 to k do
    M[1][i] = 0
  end for
  for i = 2 to n do
    for j = 2 to k do
      M[i][j] = ∞
      for x = 1 to (i - 1) do
        s = max(M[x][j - 1], deviation((x + 1), i, frequency_list))
        if (M[i][j] > s) then
          M[i][j] = s
          break_list[i][j] = x
        end if
      end for
    end for
  end for

```

até uma posição final (*end*). Já na primeira linha supõe-se que haja somente um valor de potencial na *frequency_list*, então o valor zero é armazenado em cada posição.

O restante do algoritmo preenche a matriz *M* com os valores dos desvios padrão que vão sendo calculados para cada partição, sendo que o valor da partição anterior é analisado no preenchimento das posições. Além disso, para cada partição, a posição ótima de corte é armazenada na *break_list*.

Como um exemplo, dada a amostra com uma distribuição hipotética de potenciais apresentada na Figura 6.2, ao executar o procedimento *cluster*(*frequency_list*, *k*, *break_list*) com *k* = 5, a *break_list* apresentada na Tabela 6.1 será retornada.

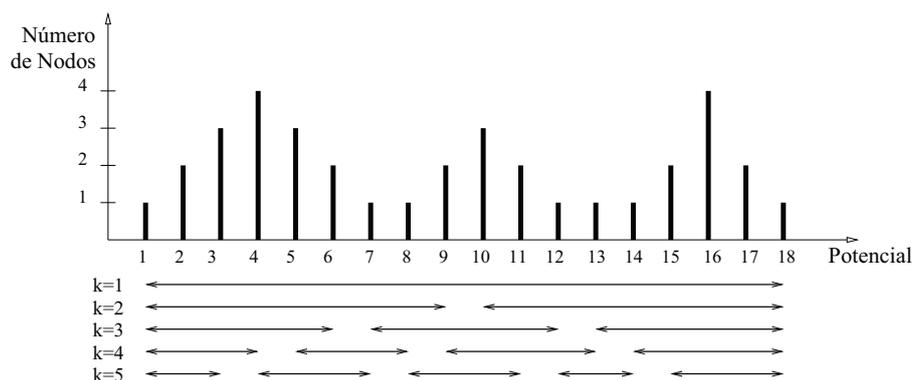


Figura 6.2: Distribuição hipotética de potenciais.

A partir de um processo de “trás para frente” na *break_list*, obtém-se os valores de corte para o *k* especificado. Para tanto, obtém-se, primeiramente, a posição do último corte (o mais à direita). Este valor é obtido a partir da última posição da *break_list*, ou seja, *break_list*[18, 5], cujo valor é 14. A seguir, obtém-se o valor que está na posição *break_list*[14, 4], ou seja, o valor do corte também indica a linha onde está o corte anterior (*k* - 1). A partir da posição *break_list*[14, 4]

Tabela 6.1: Exemplo de *break_list* para k igual a 5.

n	k				
	1	2	3	4	5
1	0	0	0	0	0
2	0	1	1	1	1
3	0	1	2	2	2
4	0	2	2	3	3
5	0	2	3	4	4
6	0	3	4	4	5
7	0	3	5	5	5
8	0	4	6	6	6
9	0	5	6	7	7
10	0	6	7	7	8
11	0	6	7	8	8
12	0	6	8	9	9
13	0	7	8	10	11
14	0	7	10	11	12
15	0	8	11	12	12
16	0	9	11	12	13
17	0	9	12	12	13
18	0	9	12	13	14

obtém-se a posição de corte 11. Na posição $break_list[11,3]$, obtém-se o valor 7, indicando outra posição de corte. Por fim, na posição $break_list[7,2]$ está armazenado o valor 3, que indica a posição do primeiro corte. Dessa forma, as posições de corte para cinco grupos na distribuição de potenciais da Figura 6.2 são: 3, 7, 11 e 14.

Além das posições de corte para o $k = 5$, a Tabela 6.1 também armazena as posições de corte para um número menor de grupos. Para uma divisão em três grupos, por exemplo, tem-se as seguintes posições de cortes obtidos a partir da $break_list$ da Tabela 6.1: 6 e 12. Mais detalhes sobre o algoritmo podem ser encontrados em [Oli08] e [Ski97].

A partir da $break_list$ obtém-se as posições de corte considerando-se divisões de 1 a k grupos. Para o algoritmo utilizado no APRP-Group é passado um valor máximo de k que corresponde à metade dos valores de potenciais que o nodo armazena, de forma que grupos pequenos, com menos de dois valores de potenciais, não sejam formados.

A escolha do melhor número de grupos é realizada da seguinte forma: realiza-se o somatório dos desvios padrão de cada grupo, para $k - 1$, k e $k + 1$. Divide-se a soma dos desvios padrão dos $k - 1$ grupos pela soma dos desvios padrão dos k grupos, bem como a soma dos desvios padrão dos k grupos pela soma dos desvios padrão dos $k + 1$ grupos. Enquanto o resultado da segunda divisão for maior que a primeira, continua-se o processo, passando para $k + 1$. Isso significa que a diminuição dos desvios padrão dos grupos é significativa a cada passo. O processo termina quando, em um dado passo, o resultado da segunda divisão for menor que a primeira, indicando uma perda na diminuição dos desvios padrão dos grupos, naquele passo. Observa-se que esse mecanismo não escolhe a divisão em apenas um grupo, pois deseja-se que no mínimo dois grupos sejam formados.

Dada a amostra com a distribuição hipotética de potenciais apresentada na Figura 6.2, observa-se que os dados separam-se naturalmente em 3 grupos, ou seja, o k ideal deveria ser 3. O cálculo

usado na escolha de k está apresentado na Figura 6.3. A tabela à esquerda apresenta os valores dos desvios padrão da distribuição hipotética para a separação em até cinco grupos, bem como o valor calculado como critério de parada. Já à direita da mesma figura está apresentado o gráfico com esses últimos valores, onde seleciona-se o valor mais alto. Nesse caso, a escolha seria o valor correspondente ao $k = 3$, indicando uma divisão em três grupos.

$k=1$	$\sigma_1 = 5.28$	
$k=2$	$\sigma_{2a} = 2.30$ $\sigma_{2b} = 2.77$	$\frac{\sigma_1}{\sigma_{2a} + \sigma_{2b}} = 1.041$
$k=3$	$\sigma_{3a} = 1.47$ $\sigma_{3b} = 1.49$ $\sigma_{3c} = 1.42$	$\frac{\sigma_{2a} + \sigma_{2b}}{\sigma_{3a} + \sigma_{3b} + \sigma_{3c}} = 1.156$
$k=4$	$\sigma_{4a} = 1.05$ $\sigma_{4b} = 1.15$ $\sigma_{4c} = 1.33$ $\sigma_{4d} = 1.15$	$\frac{\sigma_{3a} + \sigma_{3b} + \sigma_{3c}}{\sigma_{4a} + \sigma_{4b} + \sigma_{4c} + \sigma_{4d}} = 0.934$
$k=5$	$\sigma_{5a} = 0.81$ $\sigma_{5b} = 1.05$ $\sigma_{5c} = 1.03$ $\sigma_{5d} = 1.00$ $\sigma_{5e} = 0.97$	$\frac{\sigma_{4a} + \sigma_{4b} + \sigma_{4c} + \sigma_{4d}}{\sigma_{5a} + \sigma_{5b} + \sigma_{5c} + \sigma_{5d} + \sigma_{5e}} = 0.962$

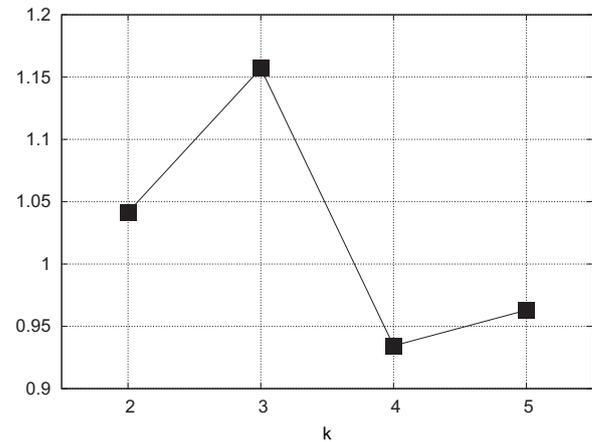


Figura 6.3: Valores dos desvios padrão se for efetuada a divisão em $k = 1, 2, 3, 4$ e 5 grupos.

6.2 Roteamento no APRP-Group

Conforme indicado nas Seções 4.4 e 5.2, os protocolos APRP e APRP-Ack geram um atraso médio na entrega das mensagens superior a outros protocolos referenciados na literatura. Há duas hipóteses para esse atraso ocorrer:

- i. em qualquer nodo, as mensagens ficam retidas até que se encontre um nodo com diferença de potencial, com relação ao seu, que justifique o repasse de uma mensagem;
- ii. em decorrência de (i), à medida que o potencial cresce, decresce a probabilidade de um nodo encontrar outro nodo com a diferença de potencial desejável, aumentando o atraso.

Para contornar a primeira hipótese, sugere-se realizar uma divisão nos créditos das mensagens, ou seja, dedicar parte dos créditos para repasse com baixo atraso, independente do potencial, e outra parte segundo a estratégia do potencial. Já para contornar a segunda hipótese, sugere-se utilizar a técnica de repasse baseada em grupos.

Grupos de Potenciais Na estratégia baseada em grupos, ao invés do repasse ser feito considerando o valor de potencial individual de cada nodo, como ocorre no APRP e APRP-Ack, o valor de potencial do grupo é utilizado. O repasse de uma mensagem ocorre quando ela sair do grupo atual e atingir algum grupo de potencial superior. Para nodos do grupo de maior potencial, é permitida a troca de mensagens dentro do mesmo grupo, já que segundo a percepção do nodo não existe um grupo

superior para repassar a mensagem. Essa estratégia evita que as mensagens fiquem retidas durante muito tempo em nodos com alto potencial de entrega. Os resultados obtidos com a utilização dessa estratégia estão apresentados na Seção 6.4, com a identificação APRP-Group1.

Créditos das Mensagens Os créditos das mensagens são divididos em tipos diferentes para alcançar uma maior redução no atraso médio na entrega das mesmas, sem que haja redução no percentual de mensagens entregues. Quando uma mensagem é criada, um número C de créditos é configurado, o qual é dividido em dois tipos - CP (créditos de potencial) e CA (créditos de atraso). Os créditos de entrega são configurados com $\lceil C/2 \rceil$ e os de atraso são configurados com $\lfloor C/2 \rfloor + 1$. Os créditos de atraso possuem um crédito a mais porque, ao se repassar uma mensagem, os dois tipos de crédito devem ter valores maiores que 1, caso contrário somente poderá ser feita a entrega direta. Dessa forma, garante-se que o número C de créditos foi repassado.

Ao repassar uma mensagem, $\lfloor C/2 \rfloor$ dos créditos ficam com o nodo que a recebe e $\lceil C/2 \rceil$ ficam com o nodo que está transmitindo a mensagem, semelhante ao APRP e APRP-Ack. Contudo, o tipo do crédito é transmitido juntamente com a mensagem e somente os créditos do tipo repassado serão decrementados. Os créditos do outro tipo poderão ser utilizados somente para o repasse direto ao nodo destino da mensagem.

6.3 Algoritmo

A estratégia de roteamento definida pelo APRP-Group se utiliza dos dois mecanismos descritos anteriormente. O Algoritmo 6.2 apresenta essa estratégia.

Assim como o APRP e APRP-Ack, alguns parâmetros precisam ser configurados, sendo eles:

- C : número de créditos que uma mensagem possui;
- W_p : tamanho da janela de tempo usada na atualização da lista de contatos que informa quem são os vizinhos do próprio nodo (*contact_list_p*);
- W_g : tamanho da janela de tempo usada na atualização da lista de contatos para agrupamento (*contact_list_g*).

O nodo que origina uma mensagem ou que possui mensagens a serem repassadas possui as seguintes estruturas e variáveis:

- o : identificação do próprio nodo;
- P_o : potencial de entrega do próprio nodo;
- P_{go} : potencial de grupo do próprio nodo;
- n : identificação do nodo encontrado. Essa informação é obtida do nodo encontrado no momento do contato entre os nodos;

- P_n : potencial de entrega do nodo encontrado. Essa informação é obtida do nodo encontrado no momento do contato entre os nodos;
- P_{gn} : potencial de grupo do nodo encontrado;
- P : potencial de entrega de um nodo;
- P_g : potencial de grupo de um nodo;
- k : número ideal de grupos;
- $break_list$: matriz com as posições de corte dos grupos;
- $time$: tempo em que houve o contato com o nodo encontrado;
- $contact_list_g$: lista de contatos para agrupamento, no formato $\langle n, P_n, time \rangle$. Essa lista é utilizada na identificação dos grupos de potenciais e armazena os vizinhos do próprio nodo e os vizinhos que são obtidos de outros nodos. Cada entrada na lista fica válida por um tempo, até que a janela de tempo (W_g) expire;
- $contact_list_p$: lista de contatos que informa quem são os vizinhos do próprio nodo, no formato $\langle n, P_n, time \rangle$. Essa lista é utilizada no cálculo do potencial de entrega dos nodos. Cada entrada na lista fica válida por um tempo, até que a janela de tempo (W_p) expire;
- $contact_list_n$: lista de contatos do nodo encontrado, no formato $\langle n, P_n, time \rangle$;
- m : identificação da mensagem a ser repassada ou entregue ao destino final. Todas as mensagens possuem uma identificação, necessária para diferenciar mensagens antigas de mensagens novas, mensagens duplicadas e mensagens que devem ser retiradas do *buffer*;
- $summary_vector_o$: lista com a identificação das mensagens e o número de créditos que estão sendo armazenadas pelo próprio nodo, no formato $\langle m, C \rangle$;
- $summary_vector_n$: lista com a identificação das mensagens que estão sendo armazenadas pelo nodo encontrado;
- CP : créditos de entrega;
- CA : créditos de atraso;
- $STATUS$: indica se a mensagem foi entregue (“*ACK*”) ou se ainda está em *buffer* (“*BUF*”);
- $type$: tipo de crédito.

 Algoritmo 6.2: Mecanismo de roteamento do APRP-Group.

```

procedure nodeMeetingNeighbor()
  obtain(n, Pn, summary_vector_n, contact_list_n, time)
  updateContactList(contact_list_p, contact_list_g, contact_list_n, n, Pn, time, Wp, Wg)
  updateListAckedIds(summary_vector_n, summary_vector_o)
  clusterKGroups(contact_list_g, k, break_list)
  computePo(contact_list_p, Po)
  computeGroup(Po, Pgo, k, break_list)
  if (nodeHasMessage(summary_vector_o)) then
    directDelivery(summary_vector_o, n, summary_vector_n)
    computeGroup(Pn, Pgn, k, break_list)
    if ((Pgn > Pgo) || ((Pgn == Pgo) && (isGreaterGroup(Pgo, k)))) then
      for all < m, CP, CA > in summary_vector_o do
        if ((CP > 1) and (!neighborHasMessage(m, summary_vector_n))) then
          sendMessageToNeighbor(summary_vector_o, n, m, CP, "CP")
        else
          if (CA > 1) then
            sendMessageToNeighbor(summary_vector_o, n, m, CA, "CA")
          end if
        end if
      end for
    else
      for all < m, CP, CA > in summary_vector_o do
        if ((CA > 1) and (!neighborHasMessage(m, summary_vector_n))) then
          sendMessageToNeighbor(summary_vector_o, n, m, CA, "CA")
        end if
      end for
    end if
  end if

procedure directDelivery(summary_vector_o, n, summary_vector_n)
  for all m in summary_vector_o do
    if ((destination(m) = n) and (!neighborHasMessage(m, summary_vector_n))) then
      sendDirectMessage(m, n)
      removeMessageAndUpdateList(summary_vector_o, m, "ACK")
    end if
  end for

procedure sendMessageToNeighbor(summary_vector_o, n, m, Cx, type)
  sendMessage(n, m, ⌊Cx/2⌋, type)
  updateMsgList(summary_vector_o, m, ⌈Cx/2⌉)

procedure receiveMessage(m, C, type)
  makeRoomForNewMessage(m)
  if (type = "CP") then
    storeInBuffer(summary_vector_o, m, C, 1, "BUF")
  else
    if (type = "CA") then
      storeInBuffer(summary_vector_o, m, 1, C, "BUF")
    end if
  end if

```

Algoritmo 6.3: Mecanismo de roteamento do APRP-Group - descrição dos procedimentos.

```

procedure computeGroup( $P, P_g, k, break\_list$ ) {Calcula o potencial do grupo do nodo passado por parâmetro.}
procedure computePo( $contact\_list\_p, P_o$ ) {Calcula o potencial do nodo origem.}
procedure clusterKGroups( $contact\_list\_g, k, break\_list$ ) {Gera a distribuição de frequência dos potenciais,
chama o procedimento descrito no Algoritmo 6.1 e retorna o número de grupos ideal e a lista com as posições
das divisões dos grupos.}
procedure createNewMessage( $m, summary\_vector\_o, C$ ) {Inclui a identificação de nova mensagem na lista
de mensagens, configura os tipos de créditos da mensagem ( $CP = \lceil C/2 \rceil$  e  $CA = \lfloor C/2 \rfloor + 1$ ), bem como seu
STATUS.}
procedure isGreaterGroup( $P_{go}, k$ ) {Verifica se potencial do grupo pertence ao maior grupo encontrado.}
procedure neighborHasMessage( $m, summary\_vector\_n$ ) {Verifica se a mensagem  $m$  está presente no
summary vector do nodo encontrado. Caso esteja retorna true, caso contrário retorna false.}
procedure nodeHasMessage( $summary\_vector\_o$ ) {Verifica se há mensagens na lista, retornando true ou
false.}
procedure obtain( $n, P_n, summary\_vector\_n, contact\_list\_n, time$ ) {Retorna a identificação, o potencial e o
summary vector do nodo encontrado, bem como o tempo do encontro.}
procedure removeMessageAndUpdateList( $summary\_vector\_o, m, "ACK"$ ) {Remove do buffer a mensagem
entregue ao destino e altera o STATUS dessa mensagem para "ACK" no summary vector do próprio nodo.}
procedure sendDirectMessage( $m, n$ ) {Envia mensagem para o destino.}
procedure sendMessage( $n, m, C, type$ ) {Repassa mensagem para o nodo.}
procedure storeInBuffer( $summary\_vector\_o, m, CP, CA, STATUS$ ) {Armazena mensagem recebida na
lista de mensagens.}
procedure updateContactList( $contact\_list\_p, contact\_list\_g, contact\_list\_n, n, P_n, time, W_p, W_g$ ) {Inclui a
identificação do nodo encontrado, potencial do nodo e tempo de contato em  $contact\_list\_p$  e  $contact\_list\_g$ .
Remove contatos antigos de acordo com as janelas de tempo ( $W_p$  e  $W_g$ ). Adiciona em  $contact\_list\_g$  os
contatos do nodo encontrado, válidos na janela de tempo.}
procedure updateListAckedIds( $summary\_vector\_n, summary\_vector\_o$ ) {Atualiza o summary vector do
próprio nodo com as mensagens confirmadas que estão no summary vector do nodo encontrado, além de remover
do buffer as mensagens confirmadas que foram atualizadas.}
procedure updateMsgList( $summary\_vector\_o, m, C$ ) {Atualiza a lista de mensagens dando o novo crédito
para a mensagem repassada.}

```

Sempre que uma mensagem é criada, $createNewMessage(m, summary_vector_o, C)$ é executado. Nesse procedimento, os créditos de entrega (CP) são configurados com $\lceil C/2 \rceil$ e os de atraso (CA) são configurados com $\lfloor C/2 \rfloor + 1$. Além disso, o $STATUS$ é configurado com "BUF", indicando que a mensagens está armazenada no *buffer* para ser entregue. As informações resumidas das mensagens criadas pelos nodos são mantidas no *summary vector* no seguinte formato: $\langle m, CP, CA, STATUS \rangle$.

Ao encontrar um nodo n , $nodeMeetingNeighbor()$ é executado. A identificação, o potencial, o *summary vector* e a lista de contatos do nodo encontrado são obtidos por $obtain(n, P_n, summary_vector_n, contact_list_n, time)$. O nodo encontrado é adicionado em $contact_list_p$ e em $contact_list_g$ e os contatos antigos são removidos de acordo com a respectiva janela de tempo. Além disso, adiciona-se na $contact_list_g$ do nodo os contatos da $contact_list_n$ que sejam válidos na janela. Assim a $contact_list_g$ armazena os vizinhos do próprio nodo e os vizinhos que são obtidos de outros nodos. A troca das listas de contatos entre os nodos ocorre para que haja uma convergência mais rápida no processo de identificação de grupos. A $contact_list_g$ será utilizada posteriormente na identificação dos grupos.

A sincronização da lista de mensagens entregues ao destino final, bem como a remoção das mesmas do *buffer* do nodo é realizado por $updateListAckedIds(summary_vector_n, summary_vector_o)$. As mensagens entregues possuem o *STATUS* com “ACK” no *summary vector*.

O agrupamento é realizado por $clusterKGroups(contact_list_g, k, break_list)$, que retorna o número de grupos ideal e a matriz com as posições de corte dos grupos. O potencial de entrega do nodo origem é calculado por $computeP_o(contact_list_p, P_o)$ e o potencial de seu grupo é calculado por $computeGroup(P, P_g, k, break_list)$.

Caso haja mensagens para serem entregues, o procedimento $directDelivery(summary_vector_o, n, summary_vector_n)$ identifica se o nodo encontrado é destino de alguma mensagem presente em sua lista para que seja feita a entrega direta, a remoção da mesma e a alteração de seu *STATUS* no *summary vector* do nodo. A entrega somente será realizada caso o nodo encontrado ainda não possua a mensagem a ser entregue. Além disso, calcula-se o potencial do grupo do nodo encontrado. Se esse potencial for maior que o do grupo do nodo origem ou ambos forem iguais e estiverem no grupo de maior potencial, é possível repassar as mensagens usando um dos dois tipos de créditos configurados. Caso contrário, somente os créditos de atraso são usados. O repasse somente será realizado caso o nodo encontrado ainda não possua a mensagem a ser entregue. Ao se repassar uma mensagem em $sendMessageToNeighbor(summary_vector_o, n, m, Cx, type)$, $\lfloor C/2 \rfloor$ dos créditos fica com o nodo que recebe a mensagem e $\lceil C/2 \rceil$ fica com o nodo que está transmitindo a mensagem, semelhante ao APRP e APRP-Ack. O tipo do crédito é transmitido juntamente com a mensagem e somente os créditos do tipo repassado são decrementados.

O procedimento $receiveMessage(m, C, type)$ é executado sempre que uma mensagem for recebida. Se houver espaço em *buffer*, o nodo armazena a mensagem e inclui a identificação, os créditos, bem como o *STATUS* da mensagem no *summary vector* do nodo. Sempre que o crédito de um tipo for recebido, o crédito de outro tipo pode ser utilizado apenas para entrega direta da mensagem.

6.4 Avaliação Experimental

A avaliação do APRP-Group foi realizada no mesmo simulador utilizado para avaliar o APRP e o APRP-Ack, sendo que os mesmos parâmetros foram utilizados. Experimentos com 5000 mensagens e com o padrão de mobilidade *Map Based Movement* foram conduzidos.

Devido aos resultados comentados na Seção 5.2 sobre o desempenho do APRP-Ack, esta Seção apresenta a comparação do APRP-Group com APRP-Ack e *Spray and Wait*. Os resultados apresentados do APRP-Group estão divididos em duas partes. A primeira, chamada de APRP-Group1, utiliza apenas a estratégia de repasses usando grupos. A segunda, chamada de APRP-Group2, utiliza a estratégia completa descrita na Seção 6.2, ou seja, a estratégia de repasses usando grupos e os créditos das mensagens.

Para o correto funcionamento do APRP-Group é necessária a configuração do parâmetro adicional W_g , que representa a janela de tempo para atualização da lista de contatos usada no mecanismo

de agrupamento. Este parâmetro foi configurado experimentalmente com $4 \times W_p$ (tamanho da janela de tempo usada na atualização da *contact_list_p*), pois, conforme experimentos realizados, com esse tempo obtém-se convergência na identificação distribuída dos grupos (em cada nodo).

Foram realizados experimentos com dois e três tipos de nodos. Nos experimentos com dois tipos de nodos, utilizaram-se os mesmos valores para velocidade e tamanho de *buffer* descritos na Seção 4.4. Dessa forma, utilizaram-se valores de velocidade entre [0,5, 1,0] m/s e [2,7, 13,9] m/s. Além disso, para o tamanho dos *buffers* utilizaram-se 40 Mbytes e 20 Mbytes. Já nos experimentos com três tipos de nodos, além dos dois tipos mencionados foi incluído um terceiro tipo com velocidades entre [18, 25] m/s e com capacidade de armazenamento de até 50 Mbytes. Esse terceiro tipo foi incluído de forma a analisar o desempenho da proposta baseada em grupos.

A seguir estão apresentados os resultados com os dois conjuntos de métricas, como apresentado na Seção 5.2.

6.4.1 Atraso na Entrega das Mensagens

As Figuras 6.4 e 6.5 apresentam o atraso médio na entrega das mensagens para dois e três tipos de nodos, respectivamente. A partir dessas figuras pode-se observar que o APRP-Group1 reduz o atraso em relação ao APRP-Ack, contudo uma redução mais significativa ocorre com o APRP-Group2 que se aproxima do *Spray and Wait*. Essa redução ocorre tanto para o cenário com dois tipos de nodos (Figura 6.4) quanto para o cenário com três tipos de nodos (Figura 6.5), ficando mais evidenciada nos cenários esparsos. Isso confirma as hipóteses de atraso tal como colocadas no início deste Capítulo. Como um exemplo, para os resultados apresentados na Figura 6.4b, o APRP-Group2 gerou uma redução no atraso em média de 18,17% em comparação ao APRP-Ack, e um aumento em média de 9,56% em comparação ao *Spray and Wait*.

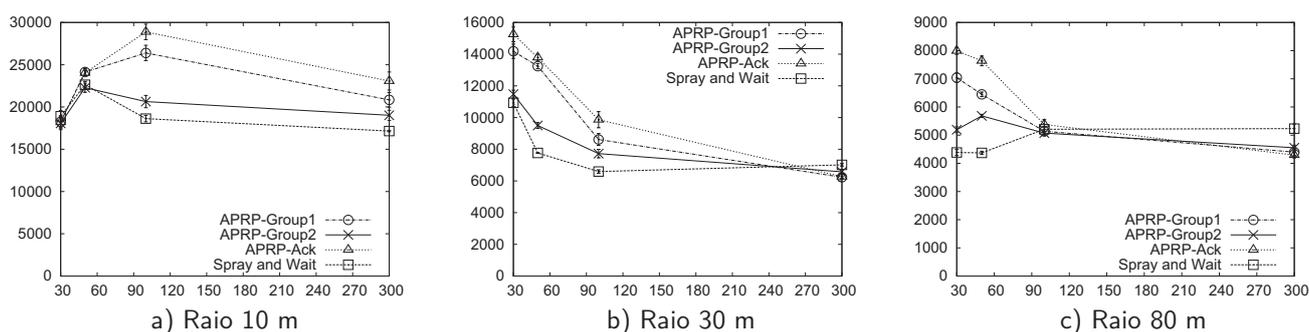


Figura 6.4: Atraso na entrega das mensagens para 2 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.2 apresenta os valores numéricos dessa figura.

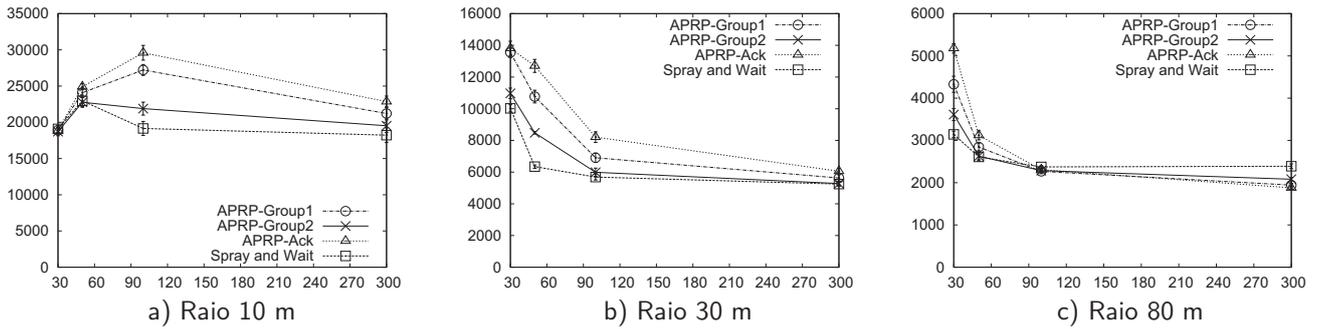


Figura 6.5: Atraso na entrega das mensagens para 3 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.3 apresenta os valores numéricos dessa figura.

6.4.2 Percentual de Mensagens Entregues

As Figuras 6.6 e 6.7 apresentam o percentual de mensagens entregues para dois e três tipos de nodos, respectivamente. Observa-se que o percentual de entrega, tanto do APRP-Group1 quanto do APRP-Group2, se mantém similar ou ligeiramente superior ao APRP-Ack na maioria dos casos, sendo que o APRP-Group2 é melhor em cenários esparsos.

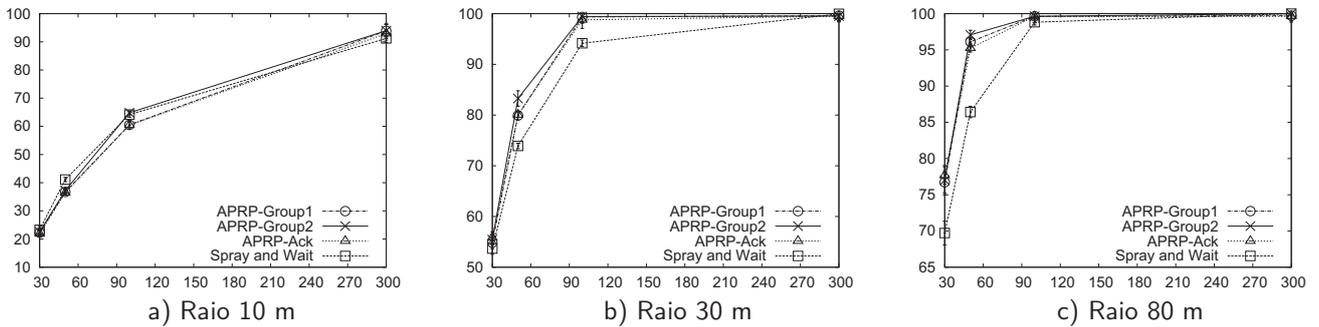


Figura 6.6: Percentual de mensagens entregues para 2 grupos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 6.4 apresenta os valores numéricos dessa figura.

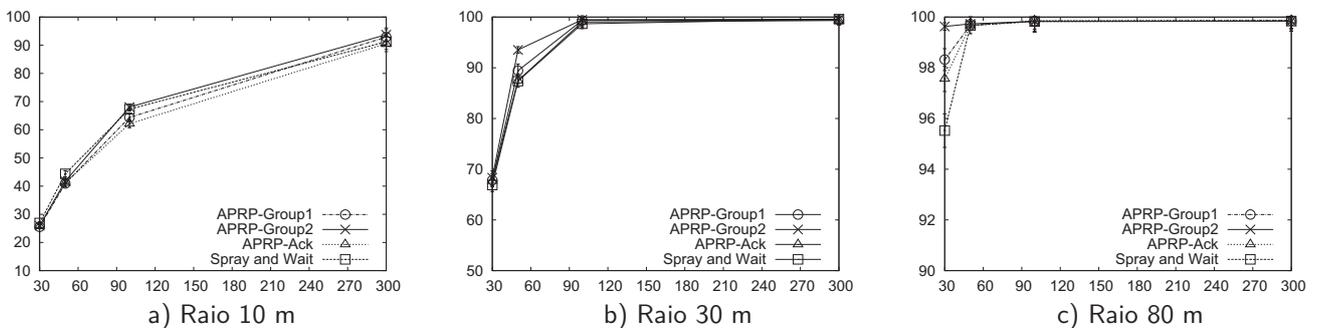


Figura 6.7: Percentual de mensagens entregues para 3 grupos. O eixo vertical representa o percentual de mensagens entregues e o horizontal representa a quantidade de nodos. A Tabela 6.5 apresenta os valores numéricos dessa figura.

Esse comportamento é devido ao APRP-Group1 e APRP-Group2 possibilitar uma maior disseminação de mensagens na rede, conforme Figura 6.8 e ainda realizar o repasse a nodos de potencial crescente. Contudo, conforme a rede se torna mais densa, seja por um número maior de nodos ou devido a um maior raio de alcance na comunicação, a probabilidade de encontrar o destino é mais alta e isso faz com que haja um maior percentual de entrega em todos os protocolos. Como um exemplo, para os resultados apresentados na Figura 6.6b, o APRP-Group2 gerou um aumento no percentual de mensagens entregues em média de 0,88% e 3,99% em comparação aos protocolos APRP-Ack e *Spray and Wait*, respectivamente.

6.4.3 Overhead

O *overhead* gerado na rede está apresentado nas Figuras 6.8 e 6.9. Apesar do *overhead* gerado tanto pelo APRP-Group1 quanto pelo APRP-Group2 ter aumentado em relação ao APRP-Ack, o mesmo continua menor que no *Spray and Wait*. O aumento do número de cópias acontece no APRP-Group1 pois os nodos de maior potencial têm a possibilidade de repassar as mensagens para nodos de mesmo grupo, não tendo que ficar à espera de um nodo com potencial próximo ao seu ou superior como no APRP-Ack. Já no APRP-Group2, uma partilha das mensagens pode ainda ser repassada a nodos de menor potencial. Apesar disso, o *overhead* se mantém muito baixo, não ocasionando sobrecarga nos recursos da rede e dos nodos. Como um exemplo, para os resultados apresentados na Figura 6.7b, o APRP-Group aumentou o *overhead* em média 44,92% em comparação ao APRP-Ack, e o reduziu em média 10,04% em comparação ao *Spray and Wait*.

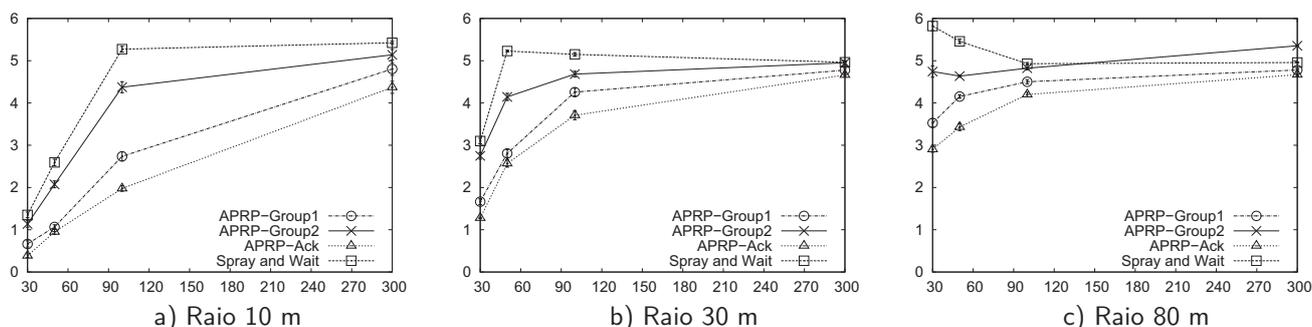


Figura 6.8: *Overhead* gerado na rede para 2 grupos. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 6.6 apresenta os valores numéricos dessa figura.

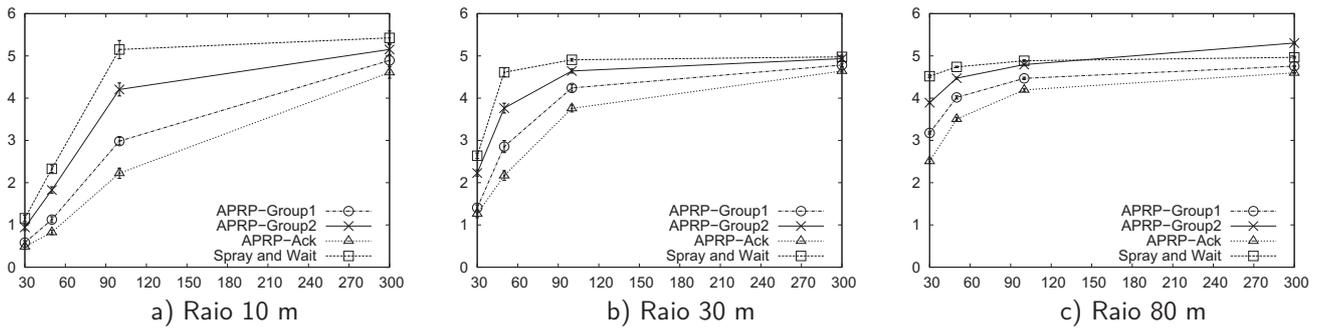


Figura 6.9: *Overhead* gerado na rede para 3 grupos. O eixo vertical representa o número de cópias e o horizontal representa a a quantidade de nodos. A Tabela 6.7 apresenta os valores numéricos dessa figura.

6.4.4 Ocupação de Buffer

As Figuras 6.10 e 6.11 apresentam a ocupação média dos *buffers* dos nodos da rede para dois e três tipos de nodos, respectivamente. Observa-se que o APRP-Ack possui uma ocupação de *buffer* menor que os demais protocolos, sendo que o APRP-Group1 se aproxima no APRP-Ack e o APRP-Group2 se aproxima do *Spray and Wait*. Esse aumento é devido a um maior número de cópias das mensagens na rede, como foi observado anteriormente.

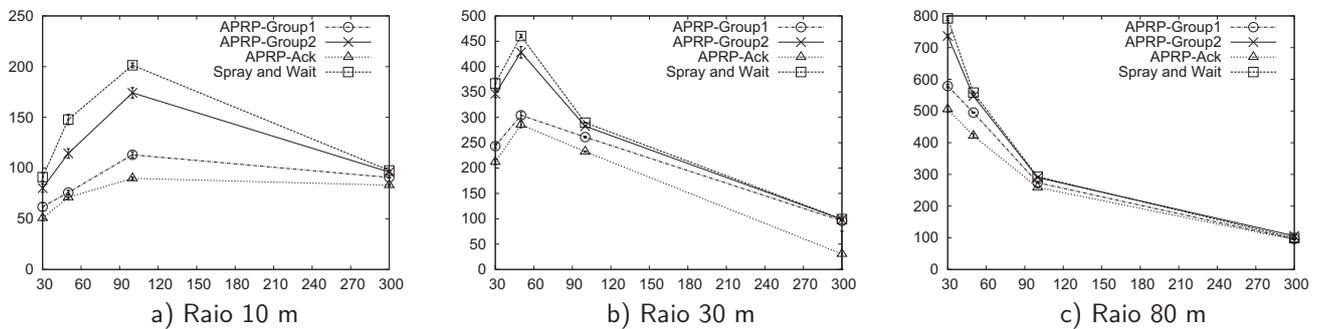


Figura 6.10: Ocupação de *buffer* para 2 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.8 apresenta os valores numéricos dessa figura.

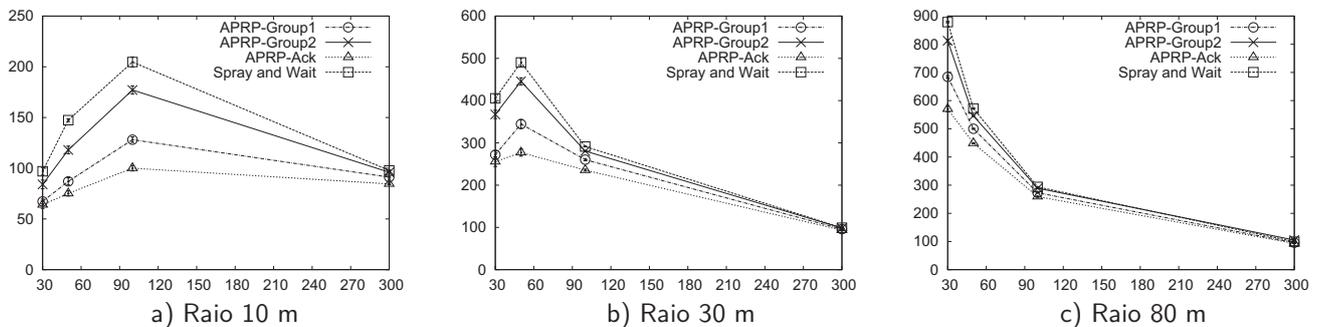


Figura 6.11: Ocupação de *buffer* para 3 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.9 apresenta os valores numéricos dessa figura.

6.4.5 Mensagens Descartadas devido a Buffer Overflow

A quantidade de mensagens descartadas devido a *buffer* cheio, para dois e três tipos de nodos, está apresentada nas Figuras 6.12 e 6.13, respectivamente.

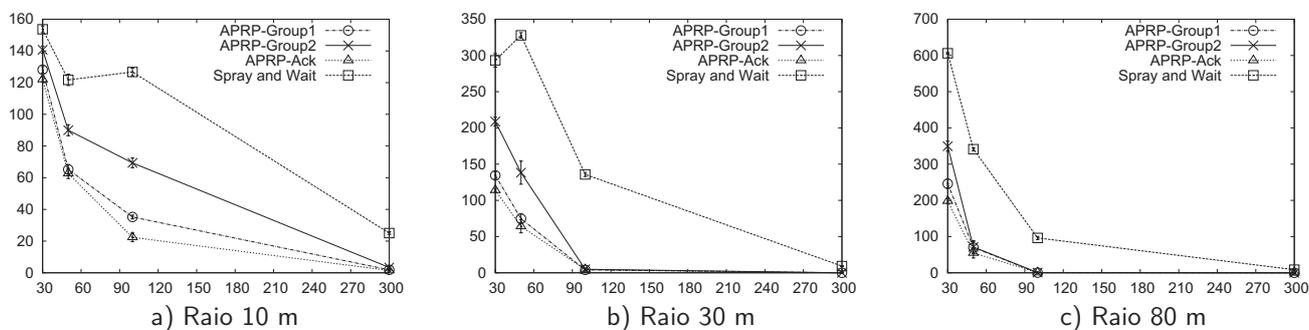


Figura 6.12: Mensagens descartadas devido a *buffer overflow* para 2 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.10 apresenta os valores numéricos dessa figura.

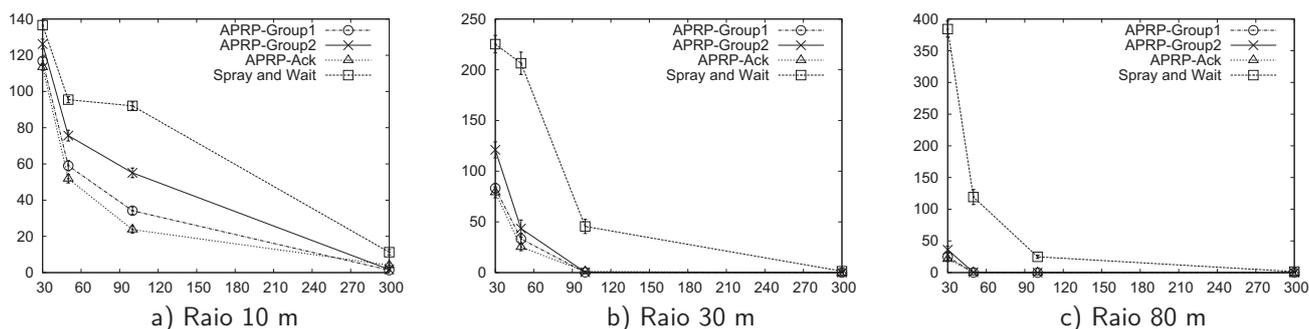


Figura 6.13: Mensagens descartadas devido a *buffer overflow* para 3 grupos. O eixo vertical representa o número de mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.11 apresenta os valores numéricos dessa figura.

Como pode-se verificar nas figuras, o *overflow* gerado pelos protocolos propostos é menor do que o gerado pelo *Spray and Wait*. Observa-se que mesmo aumentando o número de cópias das mensagens, o *overflow* gerado pelo APRP-Group2 continua baixo. Essa característica ocorre devido a utilização das mensagens de *acknowledgment*, que removem dos *buffers* as mensagens entregues.

6.4.6 Entregas Redundantes

O percentual de mensagens redundantes entregues está apresentado nas Figuras 6.14 e 6.15 para dois e três tipos de nodos, respectivamente. Através das figuras observa-se que assim como o ARP-Ack, tanto o APRP-Group1 quanto o APRP-Group2 entregam poucas mensagens depois da primeira entrega, quando comparados com o *Spray and Wait*. Verifica-se, assim, que mesmo com um aumento no número de cópias das mensagens - vide Seção 6.4.3, as entregas redundantes se mantém baixas nos protocolos propostos.

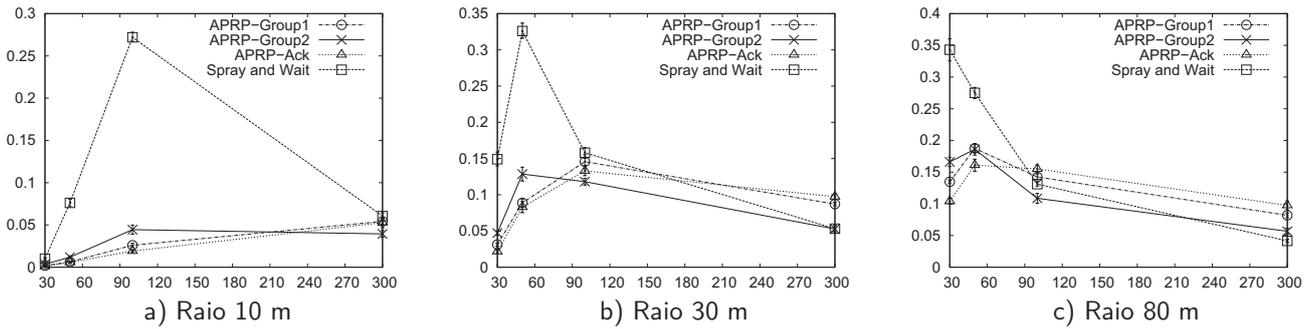


Figura 6.14: Entregas redundantes para 2 grupos. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.12 apresenta os valores numéricos dessa figura.

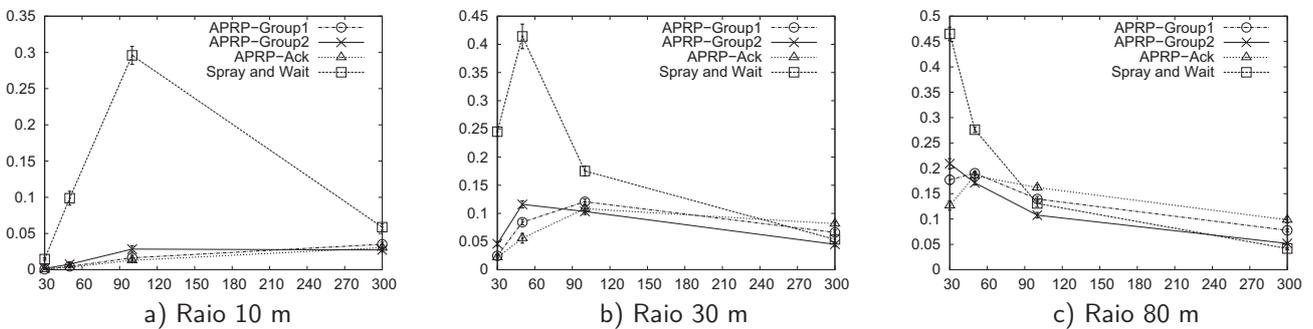


Figura 6.15: Entregas redundantes para 3 grupos. O eixo vertical representa o número de cópias das mensagens e o horizontal representa a a quantidade de nodos. A Tabela 6.13 apresenta os valores numéricos dessa figura.

6.4.7 Tempo de Vida das Mensagens

As Figuras 6.16 e 6.17 apresentam o tempo de vida das mensagens na rede para dois e três tipos de nodos, respectivamente. Os resultados mostram que o tempo de vida das mensagens com o protocolo APRP-Group2 é geralmente menor que o tempo utilizado pelos demais protocolos. Isso ocorre pois o APRP-Group2 gera um menor atraso, se comparado ao APRP-Group1 e APRP-Ack, fazendo com que as mensagens fiquem menos tempo na rede. Além disso, a utilização de *acknowledgments* também ajuda na redução desse valor.

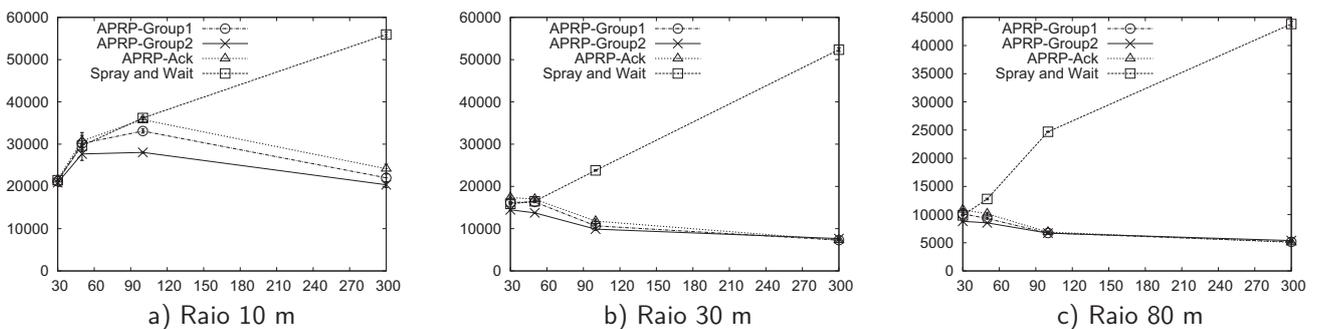


Figura 6.16: Tempo de vida das mensagens para 2 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.14 apresenta os valores numéricos dessa figura.

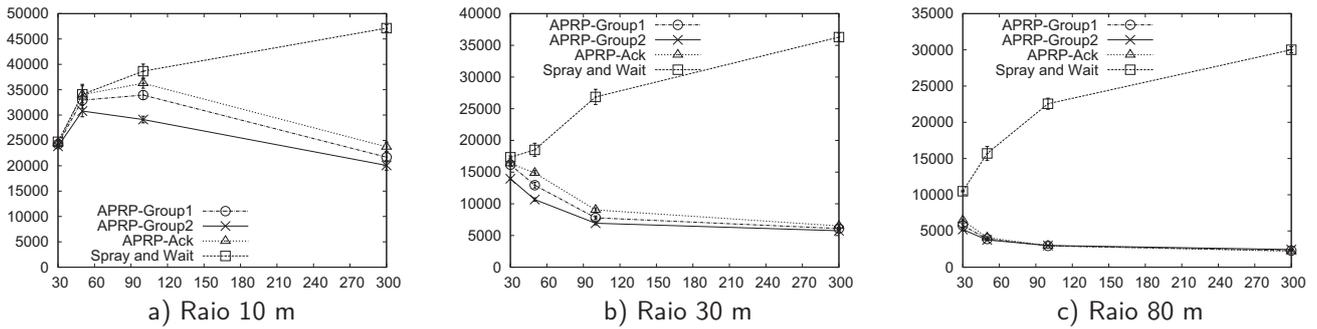


Figura 6.17: Tempo de vida das mensagens para 3 grupos. O eixo vertical representa o tempo em segundos e o horizontal representa a a quantidade de nodos. A Tabela 6.15 apresenta os valores numéricos dessa figura.

6.4.8 Tabelas com Valores dos Gráficos

Atraso na Entrega das Mensagens As Tabelas 6.2 e 6.3 apresentam os valores numéricos para o atraso na entrega das mensagens para 2 e 3 grupos, respectivamente.

Tabela 6.2: Atraso na entrega das mensagens para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	18238,46	24136,54	26389,64	20842,20
APRP-Group2	18053,90	22280,15	20642,29	19017,64
APRP-Ack	18224,24	24035,80	28882,78	23076,55
Spray and Wait	18905,78	22637,64	18625,91	17153,60
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	14185,67	13243,09	8614,84	6235,56
APRP-Group2	11475,69	9499,44	7723,28	6579,24
APRP-Ack	15251,17	13787,35	9864,25	6273,39
Spray and Wait	10927,17	7765,53	6593,73	7016,94
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	7041,59	6449,12	5139,41	4397,35
APRP-Group2	5191,95	5684,89	5073,61	4553,30
APRP-Ack	7991,94	7637,38	5380,02	4292,35
Spray and Wait	4385,104	4371,57	5209,64	5231,30

Tabela 6.3: Atraso na entrega das mensagens para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	18922,18	24086,38	27227,96	21212,74
APRP-Group2	18714,63	22766,77	21890,09	19518,76
APRP-Ack	18854,88	24908,06	29594,50	22860,64
Spray and Wait	19089,97	22968,38	19148,88	18224,61
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	13551,98	10767,51	6907,04	5629,70
APRP-Group2	10971,14	8478,63	5989,63	5279,31
APRP-Ack	13808,41	12702,73	8206,60	6056,65
Spray and Wait	10023,82	6345,31	5687,80	5245,44
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	4328,57	2844,92	2268,76	1942,53
APRP-Group2	3610,86	2627,31	2285,04	2082,04
APRP-Ack	5177,25	3120,57	2306,11	1877,61
Spray and Wait	3142,85	2608,80	2373,22	2388,24

Percentual de Mensagens Entregues As Tabelas 6.4 e 6.5 apresentam os valores numéricos para o percentual de mensagens entregues para 2 e 3 grupos, respectivamente.

Tabela 6.4: Percentual de mensagens entregues para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	22,26	36,73	60,54	93,88
APRP-Group2	22,53	37,22	64,82	93,95
APRP-Ack	21,95	36,48	60,43	92,93
Spray and Wait	23,26	41,12	64,19	91,22
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	54,84	79,91	99,34	99,60
APRP-Group2	55,45	83,27	99,43	99,53
APRP-Ack	55,83	80,06	98,77	99,51
Spray and Wait	53,68	73,91	94,16	99,97
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	76,73	96,10	99,63	99,71
APRP-Group2	77,01	97,08	99,63	100,00
APRP-Ack	77,67	95,26	99,60	99,64
Spray and Wait	69,72	86,42	98,82	99,99

Tabela 6.5: Percentual de mensagens entregues para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	25,53	40,93	64,37	92,91
APRP-Group2	25,95	41,75	68,05	93,74
APRP-Ack	25,81	41,16	62,14	90,55
Spray and Wait	26,91	44,37	67,31	91,23
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	67,83	89,43	99,35	99,43
APRP-Group2	68,31	93,53	99,48	99,59
APRP-Ack	67,71	87,49	99,03	99,33
Spray and Wait	66,88	87,37	98,68	99,60
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	98,32	99,68	99,84	99,86
APRP-Group2	99,63	99,73	99,83	99,84
APRP-Ack	97,56	99,67	99,86	99,87
Spray and Wait	95,52	99,67	99,83	99,84

Overhead As Tabelas 6.6 e 6.7 apresentam os valores numéricos para o *overhead* gerado na rede para 2 e 3 grupos, respectivamente.

Tabela 6.6: Overhead gerado na rede para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,66	1,07	2,74	4,80
APRP-Group2	1,14	2,07	4,37	5,14
APRP-Ack	0,38	0,95	1,98	4,37
Spray and Wait	1,35	2,59	5,27	5,42
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	1,66	2,81	4,26	4,78
APRP-Group2	2,75	4,14	4,68	4,95
APRP-Ack	1,28	2,57	3,71	4,66
Spray and Wait	3,09	5,22	5,15	4,96
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	3,53	4,15	4,50	4,78
APRP-Group2	4,74	4,64	4,82	5,35
APRP-Ack	2,90	3,43	4,20	4,67
Spray and Wait	5,82	5,45	4,92	4,95

Tabela 6.7: Overhead gerado na rede para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,58	1,13	2,98	4,89
APRP-Group2	0,94	1,83	4,21	5,15
APRP-Ack	0,49	0,83	2,22	4,61
Spray and Wait	1,16	2,33	5,15	5,42
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	1,41	2,85	4,24	4,78
APRP-Group2	2,23	3,76	4,64	4,93
APRP-Ack	1,26	2,17	3,76	4,64
Spray and Wait	2,64	4,61	4,91	4,98
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	3,18	4,02	4,47	4,76
APRP-Group2	3,89	4,47	4,79	5,30
APRP-Ack	2,51	3,50	4,20	4,60
Spray and Wait	4,52	4,74	4,88	4,97

Ocupação de Buffer As Tabelas 6.8 e 6.9 apresentam os valores numéricos para a ocupação de *buffer* para 2 e 3 grupos, respectivamente.

Tabela 6.8: Ocupação de *buffer* para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	61,74	75,88	113,08	90,74
APRP-Group2	80,21	114,47	174,16	96,10
APRP-Ack	50,64	71,12	89,87	83,12
Spray and Wait	91,06	147,79	201,33	97,66
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	243,37	304,17	261,03	95,88
APRP-Group2	346,47	428,32	282,49	98,69
APRP-Ack	212,01	285,50	232,43	30,49
Spray and Wait	366,81	460,36	289,55	99,36
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	578,63	495,11	274,04	96,04
APRP-Group2	736,72	547,13	290,03	105,97
APRP-Ack	505,21	421,95	258,94	94,14
Spray and Wait	792,24	558,08	292,95	99,29

Tabela 6.9: Ocupação de *buffer* para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	67,26	87,01	128,15	91,20
APRP-Group2	84,07	118,05	177,10	96,06
APRP-Ack	63,96	75,13	100,06	84,56
Spray and Wait	96,99	147,41	204,73	97,80
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	272,01	344,58	260,29	95,82
APRP-Group2	367,41	445,55	280,66	98,49
APRP-Ack	255,55	277,08	235,49	93,38
Spray and Wait	405,49	490,29	291,42	99,20
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	684,40	500,15	273,01	95,82
APRP-Group2	812,57	545,85	289,19	104,98
APRP-Ack	570,09	448,83	259,40	93,18
Spray and Wait	878,77	571,94	293,73	99,27

Mensagens Descartadas Devido a Buffer Overflow As Tabelas 6.10 e 6.11 apresentam os valores numéricos para as mensagens descartadas devido a *buffer overflow* para 2 e 3 grupos, respectivamente.

Tabela 6.10: Mensagens descartadas devido a *buffer overflow* para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	128,13	65,23	35,22	1,90
APRP-Group2	140,73	89,97	69,37	3,34
APRP-Ack	122,01	62,70	22,39	1,49
Spray and Wait	153,67	121,73	126,72	25,06
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	134,43	74,74	4,06	0,00
APRP-Group2	208,80	138,23	4,88	0,00
APRP-Ack	113,97	64,13	4,50	0,67
Spray and Wait	293,30	327,78	135,78	9,28
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	245,93	69,61	0,30	0,00
APRP-Group2	349,43	71,35	0,72	0,00
APRP-Ack	198,05	54,74	0,70	0,00
Spray and Wait	606,02	341,25	96,28	8,62

Tabela 6.11: Mensagens descartadas devido a *buffer overflow* para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	116,84	59,00	34,17	1,34
APRP-Group2	126,19	75,65	55,08	1,73
APRP-Ack	113,47	51,84	23,79	4,11
Spray and Wait	136,64	95,48	92,06	11,25
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	83,18	33,19	0,31	0,00
APRP-Group2	121,03	43,44	0,39	0,00
APRP-Ack	79,12	25,26	1,15	0,00
Spray and Wait	225,25	206,44	45,54	1,41
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	24,89	0,18	0,00	0,00
APRP-Group2	36,00	0,48	0,00	0,00
APRP-Ack	21,39	0,10	0,00	0,00
Spray and Wait	384,16	119,23	24,89	1,57

Entregas Redundantes As Tabelas 6.12 e 6.13 apresentam os valores numéricos para as mensagens que foram entregues de forma redundante para 2 e 3 grupos, respectivamente.

Tabela 6.12: Entregas redundantes para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,00	0,01	0,03	0,05
APRP-Group2	0,00	0,01	0,04	0,04
APRP-Ack	0,00	0,01	0,02	0,05
Spray and Wait	0,01	0,08	0,27	0,06
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,03	0,09	0,15	0,09
APRP-Group2	0,05	0,13	0,12	0,05
APRP-Ack	0,02	0,08	0,13	0,10
Spray and Wait	0,15	0,33	0,16	0,05
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,13	0,19	0,14	0,08
APRP-Group2	0,17	0,19	0,11	0,06
APRP-Ack	0,10	0,16	0,16	0,10
Spray and Wait	0,34	0,28	0,13	0,04

Tabela 6.13: Entregas redundantes para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,00	0,00	0,02	0,04
APRP-Group2	0,00	0,01	0,03	0,03
APRP-Ack	0,00	0,00	0,01	0,03
Spray and Wait	0,01	0,10	0,30	0,06
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,02	0,08	0,12	0,07
APRP-Group2	0,05	0,12	0,10	0,05
APRP-Ack	0,02	0,06	0,11	0,08
Spray and Wait	0,25	0,41	0,18	0,05
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	0,18	0,19	0,14	0,08
APRP-Group2	0,21	0,17	0,11	0,05
APRP-Ack	0,13	0,19	0,16	0,10
Spray and Wait	0,47	0,28	0,13	0,04

Tempo de Vida das Mensagens As Tabelas 6.14 e 6.15 apresentam os valores do tempo de vida das mensagens para 2 e 3 grupos, respectivamente.

Tabela 6.14: Tempo de vida das mensagens para 2 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	21218,49	30166,12	33103,59	22028,00
APRP-Group2	20881,14	27680,43	28044,82	20386,43
APRP-Ack	21614,36	30746,04	35788,81	24178,70
Spray and Wait	21470,12	29652,47	36180,19	55932,97
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	16205,31	16323,05	10662,75	7290,87
APRP-Group2	14456,28	13747,61	9893,48	7656,43
APRP-Ack	17370,73	16996,67	11785,11	7292,27
Spray and Wait	15861,42	16491,45	23783,33	52357,51
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	10140,16	9305,33	6726,65	5161,26
APRP-Group2	8811,21	8532,02	6704,37	5389,93
APRP-Ack	10865,00	10145,74	6902,89	5078,32
Spray and Wait	9852,22	12762,45	24696,79	43822,21

Tabela 6.15: Tempo de vida das mensagens para 3 grupos.

Raio: 10 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	24240,49	32938,23	33944,20	21702,84
APRP-Group2	23830,22	30784,67	29113,84	20067,35
APRP-Ack	24445,19	34034,68	36269,86	23736,15
Spray and Wait	24731,40	34085,85	38659,64	47137,90
Raio: 30 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	16173,55	12938,51	7793,72	6095,73
APRP-Group2	13957,32	10674,47	6937,70	5755,75
APRP-Ack	16375,20	14863,54	9039,45	6506,48
Spray and Wait	17342,90	18513,83	26858,14	36291,16
Raio: 80 m				
	30 nodos	50 nodos	100 nodos	300 nodos
APRP-Group1	5758,32	3896,91	2949,43	2283,42
APRP-Group2	5197,99	3763,35	2998,12	2444,99
APRP-Ack	6459,90	4091,74	2963,22	2215,50
Spray and Wait	10485,53	15699,13	22564,06	30017,05

6.5 Considerações Finais

Neste Capítulo foi proposta uma variação ao APRP e APRP-Ack que agrupa logicamente nodos com potenciais de entrega semelhantes e utiliza o potencial dos grupos na decisão do repasse das mensagens. Além disso, foram definidos dois tipos de créditos para ajudar na disseminação de mensagens na rede. Dessa forma, o protocolo proposto evita que mensagens fiquem retidas por muito tempo nos nodos intermediários, principalmente de potencial mais elevado, antes de encontrem o destino.

Experimentos foram realizados com tipos de grupos diferentes de nodos e observou-se, em relação ao APRP-Ack uma melhora significativa no atraso, a manutenção ou aumento no percentual de mensagens entregues, com um aumento no *overhead*. Desta forma, os objetivos do APRP-Group foram atingidos. Além disso, o APRP-Group apresenta um atraso bastante próximo ou inferior ao *Spray and Wait*, com redução no *overhead* e aumento no percentual de entrega, podendo assim ser considerado melhor que o *Spray and Wait* nos cenários analisados.

7. Conclusão

Este trabalho abordou o problema do roteamento em redes DTN estocásticas. Neste tipo de rede, as mudanças topológicas não são conhecidas a priori e nem sempre dois nodos conseguem se comunicar diretamente devido à falta de um caminho contemporâneo válido entre os nodos. Contudo, a comunicação entre eles é possível utilizando o mecanismo de *store-carry-forward*. Segundo [Cer07] e [Spy08], tais redes podem ser aplicadas em vários cenários, tais como: campos de batalha, monitoramento da vida selvagem, redes *ad hoc* veiculares, redes para fornecer acesso à Internet de baixo custo a comunidades remotas.

Devido à semelhança existente no emprego de redes DTN e redes *ad hoc*, este trabalho apresentou um estudo que ajudou a compreender como a variação da densidade da rede influencia no comportamento de entrega e atraso. Observou-se que as redes DTN são indicadas para cenários com baixa densidade, como a maior parte das configurações realizadas com até 60 nodos para raios de 20 m, 40 m, 60 m e 80 m. A utilização de DTN nesses cenários é importante devido aos altos percentuais de entrega, chegando a 100%, enquanto as redes *ad hoc* atingem até 7%, a custo de maiores atrasos.

Na tentativa de melhorar o roteamento nas redes DTN estocásticas, este trabalho propôs três variações de um protocolo de roteamento que possui como principal contribuição uma métrica chamada de “potencial de entrega”. Essa métrica mede o nível de contato entre os nodos em um intervalo de tempo e é utilizada na decisão de repasse das mensagens.

O protocolo inicialmente apresentado, o APRP, prioriza o repasse das mensagens a nodos que tenham um potencial de entrega que justifique o repasse da mesma. Os nodos com potencial de entrega mais elevado possuem um maior contato com outros nodos na rede e, portanto, têm uma probabilidade maior de encontrar o destino em um futuro próximo. Além disso, o APRP limita o número de cópias de uma mensagem. A estratégia usada é semelhante a do protocolo *Spray and Wait* com *spraying* binário. Experimentos comparando o protocolo proposto com outros protocolos apresentados na literatura foram realizados, sendo que os resultados revelaram uma considerável diminuição no *overhead*, e níveis de entrega compatíveis, principalmente, com o protocolo *Spray and Wait*. Como um exemplo, pode-se citar um cenário que utilizou o padrão de mobilidade *Map Based Movement*, com raio de alcance de 80 m, 50 nodos na área e com 5000 mensagens sendo criadas. Nesse cenário, em comparação ao protocolo *Spray and Wait*, o protocolo APRP reduziu o *overhead* em 27,8%, aumentou a probabilidade de mensagens entregues em 2,0% e gerou um aumento no atraso médio na entrega das mensagens em 54,9%. Ao considerar as nove configurações mais esparsas apresentadas na Seção 4.4, o APRP apresentou, quando comparado ao *Spray and Wait*, redução no percentual de mensagens entregues em média de 1,2%, redução no *overhead* em média de 43,2% e aumento no atraso em média de 31,8%.

Devido à estratégia de repasse usada no APRP, pode ocorrer uma maior sobrecarga nos nodos de maior potencial, conforme apresentado na Seção 4.4.6. Com o objetivo desses nodos evitarem

o descarte de mensagens importantes e melhorarem a probabilidade de entrega das mensagens, foi proposto o APRP-Ack. Esse protocolo utiliza mensagens de *acknowledgment* para remover as mensagens já entregues do *buffer* dos nodos intermediários, liberando espaço para que novas mensagens sejam armazenadas. O APRP-Ack utiliza a mesma estratégia de repasse do APRP e também limita o número de cópias das mensagens. A partir dos experimentos realizados, mostrou-se que o desempenho do APRP-Ack foi superior ao APRP e *Spray and Wait*, tanto no percentual de mensagens entregues quanto no *overhead*. Contudo, devido às características da estratégia de repasse e ao mecanismo de *acknowledgment*, que fez com que as mensagens não fossem descartadas por *buffer overflow*, as mesmas permaneceram mais tempo em *buffer* até serem entregues, aumentando o atraso na entrega. Como um exemplo, considerando a média das dezoito configurações mais esparsas apresentados na Seção 5.2, o APRP apresentou, quando comparado ao *Spray and Wait*, aumento no percentual de mensagens entregues em média de 3,0%, redução no *overhead* em média de 51,5% e aumento no atraso em média de 45,4%.

A variação proposta para solucionar esse problema foi agrupar nodos com potenciais de entrega próximos, além de definir um novo mecanismo para administração de cópias para as mensagens. Esse mecanismo dedica parte das cópias para repasse com baixo atraso, independente do potencial, e outra parte segundo o potencial dos grupos. Tal variação foi chamada de APRP-Group e, de acordo com experimentos realizados, o protocolo proposto apresentou uma melhora significativa no atraso. Comparado com o APRP-Ack, o APRP-Group apresentou, diminuição no atraso em média de 19,0%, aumento no percentual de mensagens entregues em média de 2,2% e aumento no *overhead* em média de 71,3%. Esses resultados foram obtidos a partir da média das dezoito configurações mais esparsas apresentadas na Seção 6.4, com variação da quantidade de nodos, raio e tipos de nodos presentes na área, para o padrão de mobilidade *Map Based Movement*. Apesar de ter havido um aumento no *overhead*, este não chega a prejudicar os recursos da rede pois seu valor continua inferior a outros protocolos apresentados na literatura. Considerando a média das mesmas dezoito configurações mais esparsas, o APRP-Group apresentou, comparado ao *Spray and Wait*, aumento no percentual de mensagens entregues em média de 2,3%, redução no *overhead* em média de 13,8% e aumento no atraso em média de 9,3%.

Por fim, conclui-se que tanto o protocolo APRP, quanto suas variações, podem ser muito bem utilizados no roteamento de redes DTN estocásticas. Os protocolos APRP e APRP-Ack têm a vantagem de serem simples e poderem ser aplicados a situações que se necessite de economia de armazenamento, dado o baixo número de cópias das mensagens repassado entre os nodos da rede. Além disso, tanto o APRP-Ack quanto o APRP-Group possuem um percentual de mensagens entregues geralmente superior ao *Spray and Wait* e aos demais protocolos analisados. Já o APRP-Group, é o que se mostra com o menor atraso, ficando bastante próximo ou inferior ao *Spray and Wait*, sendo apropriado para situações que não possam suportar atrasos tão elevados quanto os gerados pelo APRP e APR-Ack.

Estudos em várias áreas são importantes de serem realizados no sentido de melhorar o funcionamento e o uso das redes DTN. Acredita-se ser importante investigar questões sobre o suporte a

aplicações. Como suporte cita-se a integração entre o roteamento e as aplicações e questões como garantia de qualidade de serviço para aplicações prioritárias. Além disso, aplicações para outros cenários que não foram descritos nesta tese também precisam ser criadas. Por fim, seria interessante realizar experimentos reais, de forma que tanto protocolos de roteamento, quanto outros aspectos pudessem ser analisados.

Bibliografia

- [Abd07] Abdulla, M.; Simon, R. “The Impact of the Mobility Model on Delay Tolerant Networking Performance Analysis”. In: 40th Annual Simulation Symposium, 2007, pp. 177–184.
- [Bal07] Balasubramanian, A.; Levine, B.; Venkataramani, A. “DTN Routing as a Resource Allocation Problem”. *SIGCOMM Computer Communication Review*, vol. 37-4, Outubro 2007, pp. 373–384.
- [Ber02] Berkhin, P. “Survey Of Clustering Data Mining Techniques”, Relatório Técnico, Accrue Software, San Jose, CA, 2002, 56p.
- [Bet01] Bettstetter, C. “Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects”. *Mobile Computing and Communications Review*, vol. 5-3, Julho 2001, pp. 55–66.
- [Bet03] Bettstetter, C.; Resta, G.; Santi, P. “The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks”. *IEEE Transactions on Mobile Computing*, vol. 2-3, Jul–Set 2003, pp. 257–269.
- [Bet04] Bettstetter, C.; Hartenstein, H.; Perez-Cost, X. “Stochastic Properties of the Random Waypoint Mobility Model”. *Wireless Networks*, vol. 10-5, Outubro 2004, pp. 555–567.
- [Bre05] Brenner, L.; Fernandes, P.; Sales, A. “The Need for and the Advantages of Generalized Tensor Algebra for Kronecker Structured Representations”. *International Journal of Simulation: Systems, Science & Technology*, vol. 6-(3–4), Fevereiro 2005, pp. 52–60.
- [Bro98] Broch, J.; Maltz, D. A.; Johnson, D. B.; Hu, Y.; Jetcheva, J. “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols”. In: The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, 1998, pp. 85–97.
- [Bur03] Burleigh, S.; Hooke, A.; Torgerson, L.; Fall, K.; Cerf, V.; Durst, B.; Scott, K.; H. Weiss. “Delay-Tolerant Networking: An Approach to Interplanetary Internet”. *IEEE Communications Magazine*, vol. 41-6, Junho 2003, pp. 128–136.
- [Bur05] Burns, B.; Brock, O.; Levine, B. N. “MV Routing and Capacity Building in Disruption Tolerant Networks”. In: IEEE Infocom, 2005, pp. 398–408.
- [Bur06] Burgess, J.; Gallagher, B.; Jensen, D.; Levine, B. N. “MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks”. In: IEEE Infocom, 2006, pp. 1–11.

- [BX03] Bui-Xuan, B.; Ferreira, A.; Jarry, A. "Computing Shortest, Fastest, and Foremost Journeys in Dynamic Networks". *International Journal of Foundations of Computer Science*, vol. 14-2, Abril 2003, pp. 267–285.
- [Cam02] Camp, T.; Boleng, J.; Davies, V. "A Survey of Mobility Models for Ad Hoc Network Research". *Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2-5, Agosto 2002, pp. 483–502.
- [Cer07] Cerf, V.; Burleigh, S.; Hooke, A.; Torgerson, L.; Durst, R.; Scott, K.; Fall, K.; Weiss, H. "Delay-Tolerant Networking Architecture". Request for Comments: 4838, 2007, 35p.
- [Che07] Chen, C.; Chen, Z. "Evaluating Contacts for Routing in Highly Partitioned Mobile Networks". In: *The First International Workshop on Mobile Opportunistic Networking*, 2007, pp. 17–24.
- [Chi98] Chiang, C.-C. "Wireless Network Multicasting", Tese de Doutorado, University of California, Los Angeles, 1998, 167p.
- [Chl03] Chlamtac, I.; Conti, M.; Liu, J. J.-N. "Mobile Ad hoc Networking: Imperatives and Challenges". *Ad Hoc Networks*, vol. 1-1, Julho 2003, pp. 13–64.
- [CON09] "United States Marine Corps - Command and Control On-the-move Network Digital Over-the-horizon Relay (CONDOR)", Capturado em: <http://www.cengen.com/DEFENSE/condor.htm>, Outubro 2009.
- [Del06] Delamare, F.; Dotti, F. L.; Fernandes, P.; Nunes, C. M.; Ost, L. C. "Analytical Modeling of Random Waypoint Mobility Patterns". In: *Third ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, 2006, pp. 106–113.
- [Die09] "UMass DieselNet", Capturado em: <http://prisms.cs.umass.edu/dome/umassdieselnet>, Outubro 2009.
- [Ekm08] Ekman, F.; Keränen, A.; Karvo, J.; Ott, J. "Working Day Movement Model". In: *First ACM SIGMOBILE International Workshop on Mobility Models for Networking Research*, 2008, pp. 33–40.
- [Fal03] Fall, K. "A Delay-Tolerant Network Architecture for Challenged Internets". In: *ACM SIGCOMM*, 2003, pp. 27–34.
- [Fal09] Fall, K.; Varadhan, K. "The ns Manual", Capturado em: <http://www.isi.edu/nsnam/ns/doc>, Janeiro 2009.

- [Fer98] Fernandes, P.; Plateau, B.; Stewart, W. J. "Efficient Descriptor - Vector Multiplication in Stochastic Automata Networks". *Journal of the ACM*, vol. 45-3, Maio 1998, pp. 381–414.
- [Fer09] Fernandes, R. M. S. "Proposta de um Mecanismo de Segurança Baseado em Troca de Chaves Assimétricas para Redes Tolerantes a Atrasos e Desconexões", Dissertação de Mestrado, Universidade Federal do Rio de Janeiro/COPPE, Rio de Janeiro, Brasil, Março 2009, 138p.
- [Fir09] "First Mile Solutions", Capturado em: <http://www.firstmilesolutions.com/>, Novembro 2009.
- [Gue87] Guerin, R. A. "Channel Occupancy Time Distribution in a Cellular Radio System". *IEEE Transactions on Vehicular Technology*, vol. 36-3, Agosto 1987, pp. 89–99.
- [Haa01] Haas, Z. J.; Pearlman, M. R. "The Performance of Query Control Schemes for the Zone Routing Protocol". *IEEE/ACM Transactions on Networking*, vol. 9-4, Agosto 2001, pp. 427–438.
- [Har05] Harras, K. A.; Almeroth, K. C.; Belding-Royer, E. M. "Delay Tolerant Mobile Networks (DTMNs): Controlled Flooding in Sparse Mobile Networks". In: *Networking*, 2005, pp. 1180–1192.
- [Har09] Harras, K. A.; Almeroth, K. C. "Controlled Flooding in Disconnected Sparse Mobile Networks". *Wireless Communications and Mobile Computing*, vol. 9-1, Janeiro 2009, pp. 21–33.
- [Hua09] Huang, J.-H.; Chen, Y.-Y.; Chen, Y.-C.; Mishra, S.; Chen, L.-J. "Improving Opportunistic Data Dissemination via Selective Forwarding". In: *The 2nd IEEE International Workshop on Opportunistic Networking (WON-09)*, 2009, pp. 1–6.
- [Hyy06] Hyytiä, E.; Lassila, P.; Virtamo, J. "Spatial Node Distribution of the Random Waypoint Mobility Model with Applications". *IEEE Transactions on Mobile Computing*, vol. 5-6, Junho 2006, pp. 680–694.
- [Jai04] Jain, S.; Fall, K.; Patra, R. "Routing in a Delay Tolerant Network". In: *ACM SIGCOMM*, 2004, pp. 145–158.
- [Jar03] Jardosh, A.; Belding-Royer, E. M.; Almeroth, K. C.; Suri, S. "Towards Realistic Mobility Models for Mobile Ad hoc Networks". In: *Ninth Annual International Conference on Mobile Computing and Networking*, 2003, pp. 217–229.
- [Jar05] Jardosh, A.; Belding-Royer, E. M.; Almeroth, K. C.; Suri, S. "Real-world Environment Models for Mobile Ad hoc Networks". *IEEE Journal on Special Areas in Communications - Special Issue on Wireless Ad hoc Networks*, vol. 23-3, Março 2005, pp. 1–21.

- [Jon05] Jones, E. P. C.; Li, L.; Ward, P. A. S. "Practical Routing in Delay-Tolerant Networks". In: ACM SIGCOMM 2005 - Workshop on delay tolerant networking and related topics (WDTN-05), 2005, pp. 237–243.
- [Jua02] Juang, P.; Oki, H.; Wang, Y.; Martonosi, M.; Peh, L. S.; Rubenstein, D. "Energy-efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet". In: 10th Annual Conference on Architectural Support for Programming Languages and Operating Systems, 2002, pp. 96–107.
- [Ker07] Keränen, A.; Ott, J. "Increasing Reality for DTN Protocol Simulations", Relatório Técnico, Helsinki University of Technology, 2007, 9p.
- [Lee09] Lee, C.; Chang, D.; Shim, Y.; Choi, N.; Kwon, T.; Choi, Y. "Regional Token Based Routing for DTNs". In: The International Conference on Information Networkin, 2009, pp. 1–5.
- [Leg06] Legendre, F.; Borrel, V.; de Amorim, M. D.; Fdida, S. "Reconsidering Microscopic Mobility Modeling for Self-Organizing Networks". *IEEE Network*, vol. 20-6, Nov-Dez 2006, pp. 4–12.
- [Lin03a] Lindgren, A.; Almquist, A.; Schelén, O. "Quality of Service Schemes for IEEE 802.11 Wireless LANs: An Evaluation". *Mobile Networking and Applications*, vol. 8-3, Junho 2003, pp. 223–235.
- [Lin03b] Lindgren, A.; Doria, A.; Schelén, O. "Probabilistic Routing in Intermittently Connected Networks". *SIGMOBILE Mobile Computing and Communications Review*, vol. 7-3, Julho 2003, pp. 19–20.
- [Lin04] Lin, G.; oubir, G.; Rajaraman, R. "Mobility models for ad hoc network simulation". In: IEEE Infocom, 2004, pp. 454–463.
- [Lin09] Lindgren, A.; Doria, A.; Davies, E.; Grasic, S. "Probabilistic Routing Protocol for Intermittently Connected Networks". Internet-Draft: draft-irtf-dtnrg-prophet-04, 2009, 66p.
- [Liu05] Liu, C.; Kaiser, J. "A Survey of Mobile Ad Hoc Network Routing Protocols", Relatório Técnico, University of Magdeburg, 2005, 36p.
- [Mae05] Maeda, K.; Sato, K.; Konishi, K.; Yamasaki, A.; Uchiyama, A.; Yamaguchi, H.; Yasumoto, K.; Higashino, T. "Getting Urban Pedestrian Flow from Simple Observation: Realistic Mobility Generation in Wireless Network Simulation". In: International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems, 2005, pp. 151–158.
- [Mus05] Musolesi, M.; Hailes, S.; Mascolo, C. "Adaptive Routing for Intermittently Connected Mobile Ad Hoc Networks". In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2005, pp. 183–189.

- [Mus07] Musolesi, M.; Mascolo, C. "Designing Mobility Models Based on Social Networks Theory". *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11-3, Julho 2007, pp. 59–70.
- [N4C09] "EU-project on Delay Tolerant Network (DTN) Technology and Test Beds Approved by European Commission", Capturado em: <http://www.itu.se/press/d2958/1.38034?1=en>, Setembro 2009.
- [Nai05] Nain, P.; Towsley, D.; Liu, B.; Liu, Z. "Properties of Random Direction Models". *IEEE Infocom*, 2005, pp. 1897–1907.
- [Nil04] Nilsson, A. "Performance Analysis of Traffic Load and Node Density in Ad hoc Networks". *The Fifth European Wireless Conference*, 2004, pp. 1–7.
- [Nun09a] Nunes, C. M.; Dotti, F. L. "Uma Nova Estratégia de Roteamento para Redes Tolerantes a Atrasos". In: XXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2009, pp. 757–770.
- [Nun09b] Nunes, C. M.; Link, E.; Dotti, F. L. "Evaluating the Impact of an Acknowledgment Strategy for APRP". In: 5th International Latin American Networking Conference, 2009, pp. 77–86.
- [Nun10] Nunes, C. M.; Dotti, F. L.; Oliveira, J. B. S. de. "APRP-Group: Roteamento para Redes DTN com Repasse baseado em Agrupamento de Nós por Potencial de Entrega". In: XXVIII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2010, pp. 451–464.
- [Oli08] Oliveira, J. B. S. de. "Two Algorithms for Automatic Document Page Layout". In: ACM Symposium on Document Engineering, 2008, pp. 141–149.
- [Oli09a] Oliveira, C. T. de; Duarte, O. C. M. B. "Uma Análise da Probabilidade de Entrega de Mensagens em Redes Tolerantes a Atrasos e Desconexões". In: XXV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2009, pp. 293–305.
- [Oli09b] Oliveira, C. T. de; Moreira, M. D. D.; Rubinstein, M. G.; Costa, L. H. M. K.; Duarte, O. C. M. B. "Redes Tolerantes a Atrasos e Desconexões". In: Minicursos do XXV Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2009, pp. 203–256.
- [Oli09c] Oliveira, E. C. R. de; Albuquerque, C. V. N. "Análise do Protocolo NECTAR em Cenários com Mobilidade e Frequentes Interrupções". In: XXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, 2009, pp. 771–784.
- [Oli09d] Oliveira, E. C. R. de; Albuquerque, C. V. N. "NECTAR: A DTN Routing Protocol based on Neighborhood Contact History". In: 24th Annual ACM Symposium on Applied Computing, 2009, pp. 40–46.

- [Ott06] Ott, J.; Kutscher, D. "A Modular Access Gateway for Managing Intermittent Connectivity in Vehicular Communications". *European Transactions on Telecommunications*, vol. 17-2, Junho 2006, pp. 159–174.
- [Paz07] Pazand, B.; McDonald, C. "A Critique of Mobility Models for Wireless Network Simulation". In: 6th Annual IEEE/ACIS International Conference on Computer and Information Science, 2007, pp. 141–146.
- [Pen04] Pentland, A. S.; Fletcher, R.; Hasson, A. "DakNet: Rethinking Connectivity in Developing Nations". *Computer*, vol. 37-1, Janeiro 2004, pp. 78– 83.
- [Per94] Perkins, C. E.; Bhagwat, P. "Highly Dynamic Destination-Sequenced Distance-Vector routing (DSDV) for Mobile Computers". In: ACM SIGCOMM, 1994, pp. 234–244.
- [Per03] Perkins, C. E.; Belding-Royer, E.; Das, S. R. "Ad hoc On-Demand Distance Vector (AODV) Routing". Request for Comments: 3561, 2003, 37p.
- [Pol09] Polat, B. K.; Khan, M.; Tuli, N.; Kailay, P. "ZBMF: Zone Based Message Ferrying for Disruption Tolerant Networks". *Communications and Networking in China*, vol. 26, Fevereiro 2009, pp. 48–59.
- [Res02] Resta, G.; Santi, P. "An Analysis of the Node Spatial Distribution of the Random Way-point Mobility Model for Ad Hoc Networks". In: Workshop On Principles of Mobile Computing, 2002, pp. 44–50.
- [Ric05] Rice, J. "SeaWeb Acoustic Communication and Navigation Network". In: Underwater Acoustic Measurements: Technologies and Results., 2005, pp. 1–7.
- [San02] Santi, P.; Blough, D. M. "An Evaluation of Connectivity in Mobile Wireless Ad Hoc Networks". In: The International Conference on Dependable Systems and Networks, 2002, pp. 89–102.
- [San08] Sandulescu, G.; Nadjm-Tehrani, S. "Opportunistic DTN Routing with Window-aware Adaptive Replication". In: 4th Asian Conference on Internet Engineering, 2008, pp. 103–112.
- [Sar07] Sarkar, S. K.; Basavaraju, T. G.; Puttamadappa, C. "Ad Hoc Mobile Wireless Networks: Principles, Protocols and Applications". Boston, MA, USA: Auerbach Publications, 2007, 336p.
- [Sco05] Scott, K. "Disruption Tolerant Networking Proxies for On-the-Move Tactical Networks". In: Military Communication Conference, 2005, pp. 3226–3231.
- [Sco07] Scott, K.; Burleigh, S. "Bundle Protocol Specification". Request for Comments: 5050, 2007, 50p.

- [Sha08] Shah, S. "Performance Evaluation of Ad Hoc Routing Protocols Using NS2 Simulation". In: *Mobile and Pervasive Computing*, 2008, pp. 18–26.
- [Ski97] Skiena, S. S. "The Algorithm Design Manual". New York, NY: Springer-Verlag New York, Inc., 1997, 504p.
- [Spy08] Spyropoulos, T.; Psounis, K.; Raghavendra, C. "Efficient Routing in Intermittently Connected Mobile Networks: the Multiple-copy Case". *IEEE/ACM Transactions on Networking*, vol. 16-1, Fevereiro 2008, pp. 77–90.
- [Sri05] Sridhara, V.; Kim, J.; Bohacek, S. "Performance of Urban Mesh Networks". In: *International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, 2005, pp. 269–277.
- [Ste94] Stewart, W. J. "Introduction to the Numerical Solution of Markov Chains". Princeton University Press, 1994, 529p.
- [Ste05] Stepanov, I.; Marron, P. J.; Rothermel, K. "Mobility Modeling of Outdoor Scenarios for MANETs". In: *Annual Simulation Symposium*, 2005, pp. 312–322.
- [Tan03] Tan, K.; Zhang, Q.; Zhu, W. "Shortest Path Routing in Partially Connected Ad hoc Networks". In: *Global Telecommunications Conference*, 2003, pp. 1038 – 1042.
- [TIE09] "TIER - Technology and Infrastructure for Emerging Regions", Capturado em: <http://tier.cs.berkeley.edu/wiki/Home>, Setembro 2009.
- [Tou09] Tournoux, P. U.; Leguay, J.; Benbadis, F.; Conan, V.; de Amorim, M. D.; Whitbeck, J. "The Accordion Phenomenon: Analysis, Characterization, and Impact on DTN Routing". In: *IEEE Infocom*, 2009, pp. 1116–1124.
- [Tsu03] Tsumochi, J.; Masayma, K.; Uehara, H.; Yokoyama, M. "Impact of Mobility Metric on Routing Protocols for Mobile Ad Hoc Networks". In: *Communications, Computers and Signal Processing*, 2003, pp. 322–325.
- [Tur09] "TurtleNet", Capturado em: <http://prisms.cs.umass.edu/dome/turtlenet>, Outubro 2009.
- [Vah00] Vahdat, A.; Becker, D. "Epidemic Routing for Partially-Connected Ad Hoc Networks", Relatório Técnico, Duke University, 2000, 14p.
- [Ven06] Venkateswaran, P.; Ghosh, R.; Das, A.; Sanyal, S.; Nandi, R. "An Obstacle Based Realistic Ad-Hoc Mobility Model for Social Networks". *Journal of Networks*, vol. 1-2, June 2006, pp. 37–44.

- [War03] Warthman, F. "Delay-Tolerant Networks (DTNs): A Tutorial v1.1", Capturado em: <http://www.dtnrg.org/docs/tutorials/warthman-1.1.pdf>, March 2003.
- [Whi09] Whitbeck, J.; Conan, V. "HYMAD: Hybrid DTN-MANET Routing for Dense and Highly Dynamic Wireless Networks". In: Third International IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications, 2009, pp. 1–7.
- [Xu05] Xu, R.; II, D. W. "Survey of Clustering Algorithms". *IEEE Transactions on Neural Networks*, vol. 16-3, Maio 2005, pp. 645–678.
- [Yoo06] Yoon, J.; Noble, B. D.; Liu, M.; Kim, M. "Building Realistic Mobility Models from Coarse-grained Traces". In: The International Conference on Mobile Systems, Applications, and Services, 2006, pp. 177–190.
- [Zha06] Zhang, Z. "Routing in Intermittently Connected Mobile Ad hoc Networks and Delay Tolerant Networks: Overview and Challenges". *IEEE Communications Surveys & Tutorials*, vol. 8-1, First Quarter 2006, pp. 24–37.
- [Zha07] Zhang, X.; Kurose, J.; Levine, B. N.; Towsley, D.; Zhang, H. "Study of a Bus-based Disruption-Tolerant Network: Mobility Modeling and Impact on Routing". In: The Annual International Conference on Mobile Computing and Networking, 2007, pp. 195–206.

A. Modelagem do Random Waypoint

Este Apêndice apresenta um estudo, publicado em [Del06], no qual o padrão de mobilidade *Random Waypoint* foi modelado, através de um método analítico. Essa modelagem tem o objetivo de derivar a distribuição espacial do nodo, ou seja, a distribuição da probabilidade de um nodo estar em uma posição da área analisada. Como alguns experimentos apresentados no Capítulo 4 utilizaram o padrão de mobilidade *Random Waypoint* para definir o movimento dos nodos na rede, com esse estudo foi possível ajustar o tempo dos experimentos para atingir a distribuição espacial do nodo compatível com a obtida através do método analítico.

A modelagem dos padrões de mobilidade encontrada na literatura propõe o uso de simulação ou de modelos analíticos para a sua representação. Enquanto os modelos analíticos podem fornecer índices mais confiáveis do que os modelos de simulação, a representação de alguns aspectos de determinado sistema pode se tornar bastante complexa ou levar a modelos não tratáveis. Uma alternativa para suportar tal complexidade é o uso de formalismos Markovianos estruturados, como SAN (*Stochastic Automata Networks*) [Fer98] (utilizado neste trabalho). Esses formalismos são usados para modelar sistemas complexos e com paralelismo intrínseco. Contudo, o uso de tais formalismos obriga a consideração de um espaço de estado discreto, apesar da representação da mobilidade física de um nodo ser intuitivamente descrita como um modelo de espaço de estado contínuo.

No formalismo SAN, a ideia básica é representar um sistema como uma coleção de subsistemas com um comportamento independente (transições locais) e interdependências ocasionais (taxas funcionais e eventos sincronizantes). Cada subsistema é descrito como um autônomo estocástico, ou seja, um autômato no qual as transições são rotuladas com informações probabilísticas e de tempo. Um modelo SAN tem sempre um modelo de uma cadeia de Markov equivalente [Bre05] [Ste94]. Além disso, há dois tipos de eventos que trocam o estado de um modelo SAN: eventos locais e eventos sincronizantes. Eventos locais trocam o estado local de somente um autômato, enquanto eventos sincronizantes trocam o estado local de vários autômatos simultaneamente.

Uma das vantagens da utilização de SAN na modelagem de padrões de mobilidade se dá pelo fato de com SAN ser possível entender alguns aspectos a respeito do padrão modelado, como por exemplo, tempo de pausa, velocidade, estratégias de roteamento usadas, entre outras. Além disso, como SAN é um formalismo modular, torna-se fácil expandir o modelo inicial para modelos mais detalhados e com mais características, como por exemplo, o número de dimensões da área e o formato da área modelada, em um modelo relativamente pequeno se comparado com a sua cadeia de Markov equivalente. Este Apêndice analisa a distribuição espacial do nodo no padrão de mobilidade *Random Waypoint* usando SAN em uma rede considerando tanto uma superfície regular quanto irregular.

A.1 Modelagem de Superfícies Quadradas em SAN

Esta seção apresenta a modelagem do padrão de mobilidade *Random Waypoint* em SAN. O modelo com uma dimensão está apresentado na Figura A.1, o qual é composto por dois autômatos: *Dest_MN* e *Loc_MN*. O autômato *Dest_MN* representa o destino escolhido aleatoriamente, já o autômato *Loc_MN* representa a localização do nó na área. Ambos autômatos possuem N estados e cada estado representa uma região de M/N m. A região de M m é modelada como uma linha, onde um nó pode se mover para leste ou oeste. Para modelar tal realidade usando um formalismo de espaço discreto é necessário assumir o espaço com N slots.

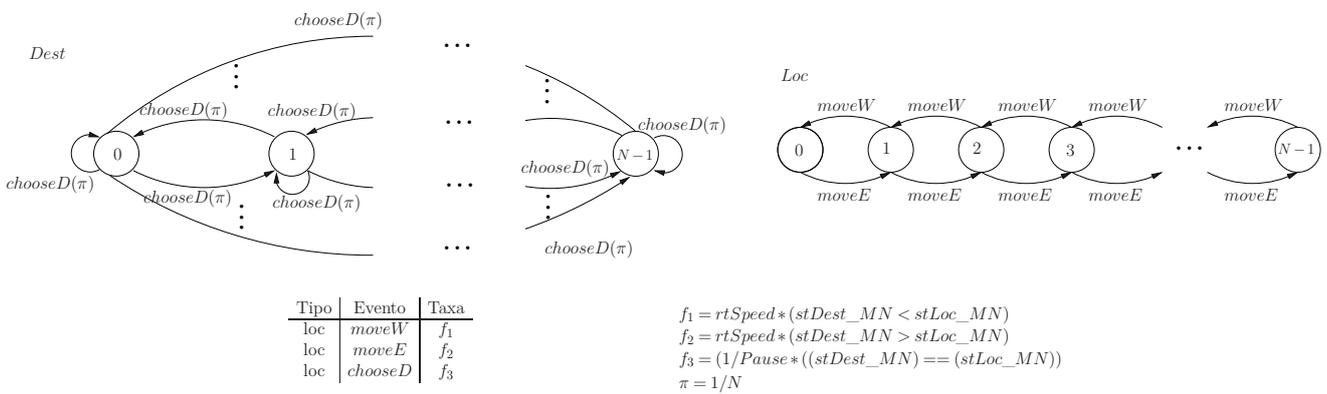


Figura A.1: Modelo SAN do *Random Waypoint* em uma dimensão.

Nesse modelo, a escolha do próximo destino é realizada pelo evento *chooseD* e a movimentação do nó para leste e oeste é feita pelos eventos *moveE* e *moveW*, respectivamente. Uma descrição mais detalhada deste modelo está apresentada em [Del06].

Para validar o modelo apresentado na Figura A.1, comparou-se os resultados alcançados com os apresentados em [Bet03] e [Res02], e verificou-se um erro muito pequeno, chegando a $10e-5$.

A Figura A.2 apresenta a distribuição de probabilidade de 20 slots para uma região de 1000 m, analisando a mobilidade do nó com uma velocidade de 32 m/s e tempos de pausa de 0,001, 4, 64 e 256 segundos. Nessa figura pode-se observar que conforme o tempo de pausa aumenta o nó se espalha mais na região, isto é, o resultado tende à distribuição uniforme. Isso é esperado pois quando o tempo de pausa aumenta, o tempo gasto no caminho se torna menos importante do que o tempo de pausa no cálculo da distribuição de probabilidade do nó. Além disso, conforme o tempo de pausa aumenta, a probabilidade tende a ser distribuída uniformemente, uma vez que o destino é escolhido equiprovavelmente.

Uma outra abordagem analisada é a distribuição do nó com diferentes valores de velocidade. De acordo com [Bet03] e [Res02], a velocidade não afeta a distribuição considerando um tempo de pausa igual a zero. Embora se possa verificar esse comportamento, observou-se também que para um tempo de pausa significativo (pausa de 60 segundos) a distribuição de probabilidade foi diferente para diferentes velocidades (1 m/s, 4 m/s, 16 m/s e 64 m/s). De fato, quanto mais relevante for o tempo de pausa em comparação com a velocidade do nó, mais próximos os resultados estarão de

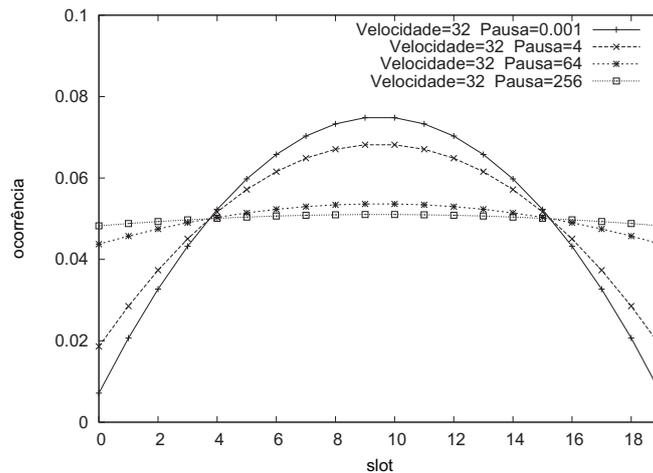


Figura A.2: Distribuição de probabilidade do nó com variação do tempo de pausa.

uma distribuição equiprovável. Esses resultados foram consistentes com as conclusões apresentadas em [Res02].

Já o modelo SAN proposto para duas dimensões representa a região modelada em um espaço bi-dimensional, no qual o nó pode se mover para norte, sul, leste e oeste. O modelo usado para representar esse movimento está apresentado na Figura A.3. Esse modelo foi obtido através de uma extensão natural do modelo de uma dimensão. A mesma estrutura do modelo de uma dimensão foi replicada, obtendo-se quatro autômatos para representar a SAN correspondente a esse novo modelo.

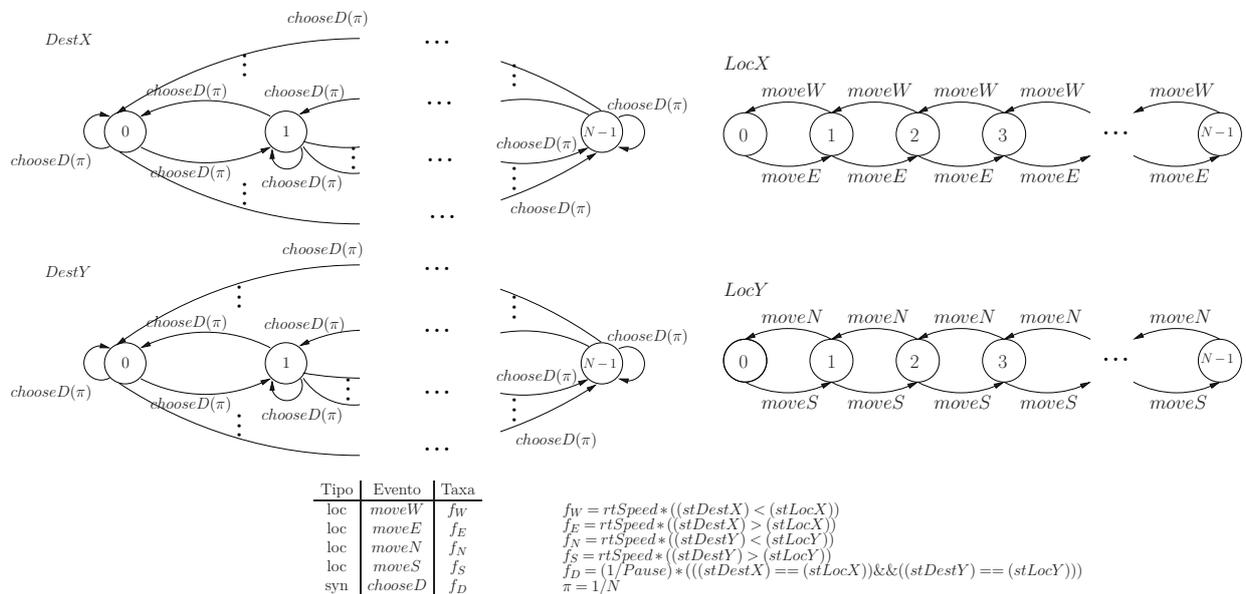


Figura A.3: Modelo SAN do *Random Waypoint* em duas dimensões.

Nesse modelo há dois autômatos para descrever a posição destino do nó (*DestX* e *DestY*) e dois autômatos para descrever a localização corrente do mesmo (*LocX* e *LocY*). Nos autômatos, um ponto é obtido por uma coordenada x e y , isto é, a superfície é dividida em $N slots \times N slots$. A escolha do próximo destino (evento *chooseD*) é representada por um evento sincronizante. A

troca do destino pode trocar ambas coordenadas x e y , representando a ocorrência de transições nos autômatos $DestX$ e $DestY$.

Além disso, o movimento é representado por quatro eventos locais que descrevem movimentos nas direções norte, sul, leste e oeste (respectivamente $moveN$, $moveS$, $moveW$, e $moveE$). Contudo, como o nodo se move em um espaço discreto, uma estratégia de roteamento foi necessária para que o nodo saísse de um ponto e chegasse a outro. A estratégia que melhor se aproximou do comportamento do movimento contínuo foi a chamada de “glutão”. Essa estratégia permite um nodo se mover somente se esse movimento é na direção mais distante em relação ao destino. Uma descrição mais detalhada desse modelo encontra-se em [Del06].

O uso de uma área de duas dimensões apresenta resultados semelhantes aos encontrados nos modelos com uma dimensão. A Figura A.4 apresenta a distribuição espacial do nodo para uma superfície de $1.000\text{ m} \times 1.000\text{ m}$ que foi dividida em $20\text{ slots} \times 20\text{ slots}$, com alteração do tempo de pausa e com a velocidade fixa em 32 m/s .

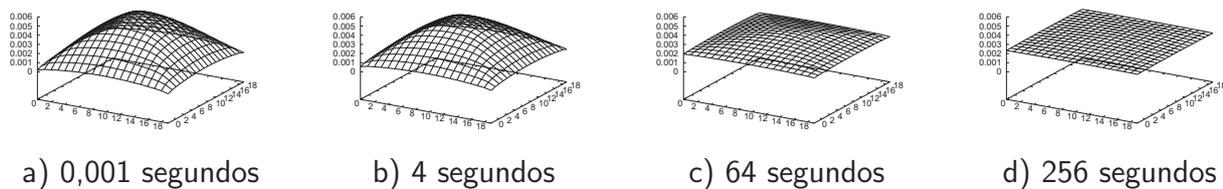


Figura A.4: Distribuição espacial do nodo com variação do tempo de pausa e velocidade de 32 m/s .

Os resultados encontrados, quando não se tem pausa, foram compatíveis com estudos existentes em [Bet03] e [Res02], que consideram um espaço de estados contínuos. Observou-se que a velocidade do nodo não altera sua distribuição espacial quando não se tem tempo de pausa. Além disso, constatou-se que essa distribuição espacial não é uniforme, sendo que os nodos se concentram mais no centro da área do que nas bordas. Esse mesmo resultado foi obtido quando o tempo de pausa foi fixado em 60 segundos e a velocidade do nodo variou em 1 m/s , 4 m/s , 16 m/s e 64 m/s [Del06].

Estudos mostraram que embora a posição inicial do nodo venha de uma distribuição uniforme, o padrão de mobilidade altera essa distribuição durante o movimento. Esse efeito ocorre porque os nodos tendem a atravessar o centro da região modelada com uma frequência relativamente maior e, sem um tempo de pausa, o padrão é independente da escolha da velocidade do nodo. Contudo, à medida que o tempo de pausa aumenta, a distribuição se aproxima da distribuição uniforme.

Além da extensão do modelo de uma dimensão em duas dimensões, pode-se estendê-lo novamente para modelar áreas com três dimensões. Essa extensão é possível adicionando mais autômatos e também mais complexidade nas funções. Uma modelagem experimental de três dimensões com o *Random Waypoint* foi realizada, sendo incluídos mais dois autômatos no modelo de duas dimensões.

Esse experimento considerou $8\text{ slots} \times 8\text{ slots} \times 8\text{ slots}$ descrevendo um espaço de 1 km^3 , um tempo de pausa de $0,001$ segundos e velocidade de 10 m/s . A Figura A.5 representa a distribuição espacial do nodo para algumas fatias no eixo z . Conforme esperado, há uma alta probabilidade do nodo estar no centro da região.

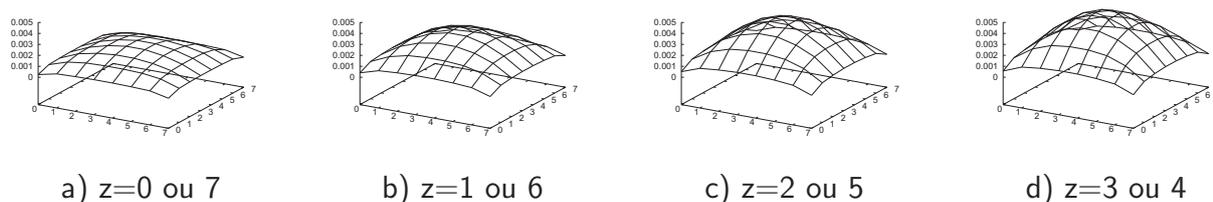


Figura A.5: Distribuição espacial do nodo em três dimensões.

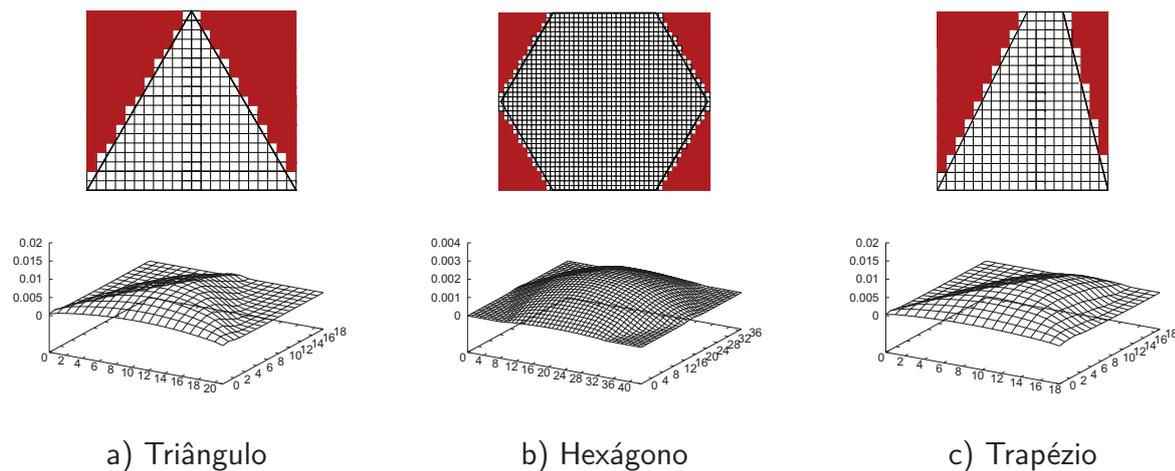


Figura A.6: Random Waypoint na superfície regular e irregular - distribuição espacial do nodo.

Além disso, como trabalho futuro, é possível estender o modelo para áreas estruturadas, como prédios com obstáculos, por exemplo. Os obstáculos podem ser modelados em SAN através de taxas funcionais assumidas para as transições e através de funções de alcançabilidade que limitam os estados alcançáveis.

A.2 Modelagem de Superfícies Irregulares em SAN

Uma importante característica de SAN é a construção modular que ajuda o reuso e a atualização de partes do modelo. Esta seção apresenta uma extensão realizada no modelo de duas dimensões apresentado na seção anterior para modelar áreas convexas, tais como triângulos, hexágonos (como em [Hyy06]) e polígonos irregulares. Os resultados obtidos de todos os modelos com superfícies irregulares mostraram consistência com aqueles obtidos previamente para superfícies quadradas.

Para modelar áreas irregulares, foi considerado o modelo utilizado nas superfícies quadradas e foram adicionados obstáculos nas bordas, para o movimento não ocorrer nessas regiões. Nas Figuras A.6a, A.6b e A.6c essas regiões estão representadas em preto, enquanto as regiões alcançáveis estão em branco. Devido à discretização da superfície, os *slots* alcançáveis são uma aproximação do polígono esperado.

A estratégia de roteamento usada é uma variação do algoritmo glutão, sendo feita uma pequena mudança para prevenir o bloqueamento que pode acontecer, em alguns casos, perto de uma borda irregular.

As regiões modeladas estão apresentadas na parte de cima da Figura A.6 e suas correspondentes distribuições espaciais estão na parte de baixo da mesma figura.

Para representar um triângulo, foram assumidos 22 *slots* como base e lados, e 19 *slots* como altura. Os resultados da distribuição espacial do nodo dessa área foram obtidos considerando cada *slot* com 50 m, como na área quadrada, com velocidade de 32 m/s e tempo de pausa de 0,001 segundos.

Alguns experimentos foram feitos para a superfície triangular, fixando a velocidade do nodo em 32 m/s e variando o tempo de pausa em 0,001, 4, 64 e 256 segundos. A Figura A.7 mostra a distribuição de probabilidade de uma seção feita na metade do eixo *y* do triângulo. Experimentos semelhantes foram feitos para o tempo de pausa fixo em 60 segundos e a velocidade variando em 1 m/s, 4 m/s, 16 m/s e 64 m/s. Os resultados concluíram que a proporção entre os valores de pausa e velocidade parecem não serem importantes.

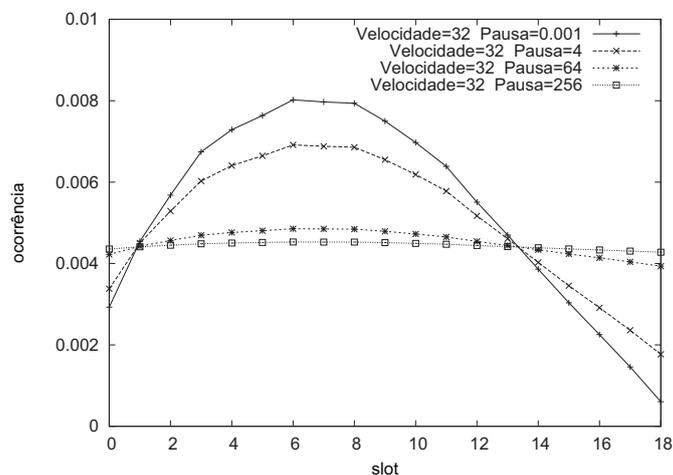


Figura A.7: Corte feito na metade do eixo *y* da superfície do triângulo - variando a pausa.

Adicionalmente, como validação das modelagens, pode-se observar que a curva representando a distribuição do nodo para a metade da seção do eixo *y* com velocidade de 32 m/s e tempo de pausa de 0,001 segundos apresenta um comportamento muito parecido com os resultados apresentados em [Hyy06].

Na área hexagonal, foram consideradas as dimensões do triângulo para cada uma das seis partes que compõem o hexágono. Então a área hexagonal foi descrita dentro de um retângulo de 44 *slots* × 38 *slots*. O resultado da distribuição espacial do nodo apresentado na Figura A.6b foi obtido com velocidade de 32 m/s e tempo de pausa de 0,001 segundos.

Esses mesmos valores de velocidade e pausa foram considerados nos resultados do trapézio (Figura A.6c), que foram descritos com tamanho de 19 *slots* × 20 *slots*. Esse último exemplo pretende mostrar que superfícies irregulares também podem ser modeladas em SAN.

Além disso, é importante observar que a modelagem de outras áreas se torna elegante e natural, precisando principalmente especializar a definição das transições do movimento e escolha do destino, mas mantendo genérico as partes principais do modelo de duas dimensões. Portanto, fica clara a

possibilidade de modelar áreas mais realistas para fazer predições sobre a distribuição espacial do nodo dessas áreas.

De fato, mesmo polígonos não convexos poderiam ser modelados, mas para esses casos é necessária uma definição mais cuidadosa das funções da estratégia de roteamento.

Analogamente, essa mesma análise pode ser realizada para outros padrões de mobilidade, como o *Random Direction*, por exemplo.