

PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

**Gerenciamento de *Buffers* em Redes DTN baseado em Entropia
de Mobilidade**

Polianny Almeida Lima

Manaus – Amazonas
2013

Polianny Almeida Lima

**Gerenciamento de *Buffers* em Redes DTN baseado em Entropia
de Mobilidade**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática do Instituto de Computação da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Informática.

Área de concentração: Redes de Computadores.

Orientador: Prof. Dr.-Ing. Edjair de Souza Mota

Polianny Almeida Lima

Gerenciamento de *Buffers* em Redes DTN baseado em Entropia de Mobilidade

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática do Instituto de Computação da Universidade Federal do Amazonas, como requisito parcial para obtenção do título de Mestre em Informática.

Área de concentração: Redes de Computadores.

Banca Examinadora

Prof. Dr.-Ing. Edjair de Souza Mota
Universidade Federal do Amazonas

Prof. Dr. Leandro Silva Galvão de Carvalho
Universidade Federal do Amazonas

Prof. Dr. Edson Nascimento Silva Júnior
Universidade Federal do Amazonas

Prof. Dr. Celso Barbosa Carvalho
Universidade Federal do Amazonas

Manaus – Amazonas

2013

Aos meus pais e meus avós, especialmente meu Avó Zacarias Marcolino de Almeida (in memoriam), pela criação e princípios que me ensinaram e por acreditarem e contribuírem em minha formação acadêmica.

A minha família e amigos pelo amor e carinho nesses anos de estudo.

Agradecimentos

Não poderia entregar esse trabalho ganhando todos os méritos (e as críticas...) sem reconhecer o esforço de outras pessoas para que ele fosse realizado.

Em primeiro lugar, agradeço a Deus pelo dom da vida e responsável por toda inspiração. O que seria de nós sem a fé que Nele temos.

Em segundo lugar, preciso agradecer ao meu orientador Professor Edjair Mota, que gentilmente aceitou o desafio de me acompanhar nessa árdua empreitada de três anos de mestrado.

Em seguida, ao Professor JB, que tão prontamente respondeu à minha solicitação de ajuda e acreditou na realização deste trabalho muitas vezes mais do que eu. Muito obrigado.

Aos meus amigos do GRCM pela amizade, compreensão e aprendizado compartilhado. Por me aturarem todos esses anos e fazerem parte desta família que ganhei em Manaus.

Ao PPGI e Professores deste Instituto de Computação pela oportunidade e garra com que vêm coordenando este curso. Valeu pela amizade.

Aos meus pais e toda família pela colaboração, dedicação e amor nesses anos de estudo.

Aos amigos pelo carinho, companheirismo que me ajudaram e me orientaram nessa jornada distante da minha família.

A CAPES pelo apoio financeiro.

E a todos que fizeram deste sonho uma grande conquista.

A vocês meus sinceros agradecimentos.

Resumo

Tecnologias de redes tradicionais baseiam-se em suposições típicas, como a existência de conectividade fim a fim entre origem e destino durante todo o período de comunicação, baixas taxas de erros, mecanismos de retransmissão para reparar erros, e atrasos de comunicação relativamente pequenos. Caso uma ou mais dessas suposições não sejam alcançadas na prática, ocorrem intermitências na comunicação, criando desafios que devem ser superados a fim de garantir o sucesso na transmissão. Nesse contexto, surgiram as redes tolerantes a atraso e desconexão (DTN - *Delay Tolerant Networks*), uma tecnologia emergente para comunicações quando não há infraestrutura de rede e desconexões podem ocorrer com frequência devido à mobilidade dos nodos. Para contornar o problema das desconexões, as redes DTN utilizam o conceito de armazenamento persistente, o que se torna um problema, pois os nodos tem capacidade de *buffers* limitada. A questão é selecionar as mensagens que ficarão nos *buffers* por mais tempo levando em consideração a mobilidade imprevisível dos nodos, contatos de curta duração entre pares de nodos, armazenamento finito realizado pelos nodos e assim por diante. Este trabalho apresenta uma proposta de gerenciamento de *buffer* que transmite mensagens de forma adaptativa baseada na variação de entropia de mobilidade da rede.

Palavras-chave: Redes Oportunistas, Gerenciamento de *Buffer*, Entropia de Mobilidade, DTN (*Delay Tolerant Networks*).

Abstract

Standard networking technologies are based on typical assumptions such as the existence of connectivity end-to-end between source and destination for the entire period of communication, low error rates, retransmission mechanisms to repair errors, and relatively small communication delays. If one or more of these assumptions are not achieved in practice, blinks occur in communication, creating challenges that must be overcome to ensure the successful transmission. In this context, came the delay-tolerant networks and disconnection (DTN - Delay Tolerant Networks), an emerging technology for communications when there is no network infrastructure and disconnections may occur frequently due to the mobility of nodes. To bypass the problem of disconnections, DTN networks use the concept of persistent storage, which becomes a problem because the nodes have limited capacity of the buffers. The question is how to select the messages that remain in buffers longer considering the unpredictable mobility of nodes, short contacts between pairs of nodes, storage nodes achieved by finite and so on. This paper proposes a buffer management that transmits messages adaptively based on entropy variation of mobility network.

Keywords: Opportunistic Networks, Buffer Management, Entropy Mobility, DTN (Delay Tolerant Networks).

Sumário

1. Introdução	1
1.1 Objetivo	3
1.2 Objetivos específicos.....	3
1.3 Organização do trabalho.....	4
2. Fundamentação Teórica	4
2.1 Conceitos Básicos	4
2.1.1 Protocolos de Roteamento.....	5
2.1.2 Transferência de Custódia	7
2.1.3 Tempo de vida da mensagem (TTL)	8
2.1.4 Número de replicações	8
2.1.5 Notificação de recepção de mensagem	8
2.1.6 Classes de prioridades	9
2.2 Modelos de Mobilidade.....	9
2.3 Entropia	10
2.4 Trabalhos correlatos	12
3. Descrição do Problema.....	15
4. Metodologia.....	19
4.1 Simulador <i>The ONE</i>	19
4.2 Projeto de experimentos	21
5. Resultados	28
5.1 Taxa de ocupação dos <i>buffers</i>	28
5.2 Taxa de entrega de mensagens	30
5.3 Sobrecarga da rede	32

6. Conclusão e Trabalhos Futuros	36
Referências Bibliográficas	36

Lista de Figuras

Figura 1 Comunicação Oportunistica (Fonte[3])	2
Figura 2 Custódias da mensagem M até entrega no instante 5	8
Figura 3 Mobilidade dos nodos para envio de mensagem ao destinatário	9
Figura 4 Modelo proposto (Fonte: [17]).	13
Figura 5 <i>Buffer</i> com classe de prioridade (Fonte: [21]).	15
Figura 6 <i>Interface</i> gráfica do simulador <i>The ONE</i>	23
Figura 7 Disposição dos pontos visitados	24
Figura 8 Relação entre replicações e entropia.....	28
Figura 9 Cenário NCSU para Ocupação de <i>Buffer</i>	30
Figura 10 Cenário KAIST para Ocupação de <i>Buffer</i>	31
Figura 11 Cenário NCSU para Probabilidade de Entrega.....	32
Figura 12 Cenário KAIST para Probabilidade de Entrega.....	33
Figura 13 Cenário NCSU para Sobrecarga.	34
Figura 14 Cenário KAIST para Sobrecarga.	34

Lista de Tabelas

Tabela 1 Características das simulações.	25
Tabela 2 Cenário avaliado para encontrar o tamanho de k.	26
Tabela 3 Resultados das simulações.	26
Tabela 4 Número de Replicações.	27

Capítulo 1

Introdução

Com a popularização dos dispositivos móveis, cresceu a quantidade de usuários que, cada vez mais exigentes, desejam estar conectados à Internet a qualquer instante e em qualquer lugar. Logo, a mobilidade, um fator que rege as comunicações atuais, expõe os nodos móveis a desconexões frequentes, resultando na perda de qualidade das comunicações.

Nesse contexto, onde as modelagens tradicionais para redes com conexões persistentes e nodos estáticos são ineficientes, o conceito de Redes Tolerantes a Atrasos e Desconexões (DTN – *Delay and Disruption Tolerant Networking*) surge como uma solução para cenários de comunicação intermitente. DTN é uma tecnologia para comunicações quando não há infraestrutura de rede e desconexões podem ocorrer com frequência devido à mobilidade dos nodos [1].

Em uma DTN, o tempo de envio da mensagem pode variar substancialmente, de alguns minutos até muitas horas, dependendo de como um nodo se move, quão densa é a população e quão distante está o transmissor do receptor. Essa transferência é realizada durante um contato que pode ser de longa duração, podendo chegar a alguns dias; e depende diretamente da mobilidade dos nodos na rede. Para contornar o problema das desconexões, as DTNs utilizam o conceito de armazena-e-encaminha (*store-and-forward*) e armazenamento persistente. No entanto, essas soluções acarretam em outro problema, pois os nodos têm capacidade de armazenamento (*buffers*) limitada.

Com a utilização de armazenamento persistente, o nodo DTN é capaz de superar as limitações de conectividade intermitente e os dados podem ser retransmitidos até que sejam recebidos com sucesso em um próximo nodo. O desempenho da comunicação depende do roteamento, do algoritmo de retransmissão e de como isso reflete a realidade nos nodos. O desempenho de DTN é dependente da mobilidade e características subjacentes do nodo, explorando a mobilidade dos próprios nodos na tentativa de entregar a mensagem o destino [2].

O roteamento é caracterizado pela forma de trocar as mensagens a cada contato. Quando um nodo comunica-se com outro, eles verificam todas as mensagens armazenadas em seus *buffers* e, então, trocam as que não têm em comum. Assim, quando uma mensagem precisa ser enviada, primeiramente ela é recebida na íntegra e armazenada, para depois, quando houver um contato, ser encaminhada ao próximo nodo, que poderá ou não ser o destino.

Conforme ilustrado na Figura 1, os nodos intermediários armazenam a mensagem e a repassam no momento em que houver conexão, até que alcance o destino. Logo, não é necessário um caminho fim a fim entre origem e destino para que a mensagem seja entregue.

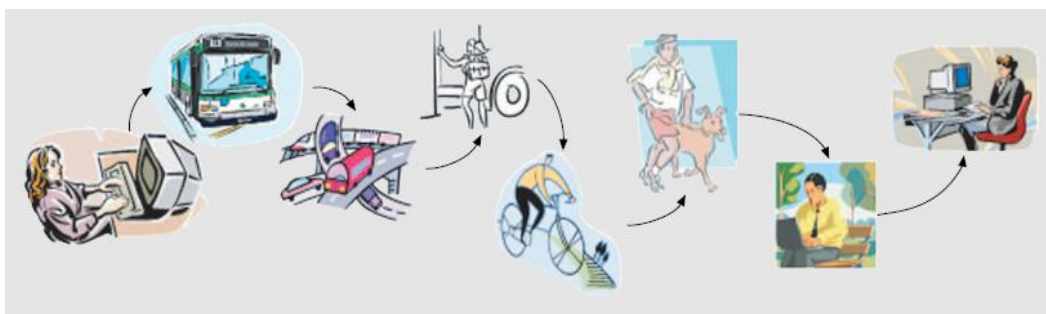


Figura 1 Comunicação Oportunística (Fonte[3]).

Um dos primeiros e mais populares algoritmos de roteamento estocástico que apresenta maior taxa de entrega é o protocolo Epidêmico, proposto por Vahdat e Becker [3]. Esse protocolo dissemina réplicas de uma mensagem entre vizinhos que estão no mesmo alcance de transmissão. Assim, a mensagem é distribuída rapidamente em todos os nodos alcançáveis na rede. Logo, quanto maior a quantidade de replicações de uma mensagem encaminhada, maior será a probabilidade de a mesma ser entregue e, menor será o atraso. Assim, mensagens vão se espalhar como uma epidemia através da rede, consumindo ao máximo os recursos disponíveis.

Algoritmos epidêmicos se mostram eficientes na entrega da mensagem, porém há um alto custo em termos do número de retransmissões (*overhead*) e consumo dos recursos de armazenamento dos nodos (*buffer*) [4]. Este desperdício não só consome os recursos de largura de banda e de *buffer*, como leva ao congestionamento da rede, provocado pelo excesso de mensagens e limitação da capacidade de *buffer*. Para aumentar a taxa de entrega em DTN, cada nodo deve levar tantas mensagens quanto possível. Ou seja, para garantir uma eventual entrega de todas as mensagens seria necessária uma capacidade de armazenamento

equivalente ao número de mensagens existentes na rede. Mas como a taxa de entrega não se limita unicamente ao tamanho dos *buffers* e tê-los em capacidade ilimitada é algo fora da realidade, os nodos não raramente terão sua capacidade de armazenamento ultrapassada e, então, surge uma pergunta: como conciliar as mensagens já armazenadas com a demanda de novas mensagens que chegam?

Baseado nesse contexto busca-se gerenciar o espaço ocupado pelas mensagens de forma a evitar o descarte e o aumento do atraso total que comprometem a taxa de entrega, o que gera congestionamento na rede e o não repasse das mensagens. Vários fatores influenciam esse comportamento, as técnicas precisam aliar recursos disponíveis, protocolos de roteamento e políticas de descarte de mensagem dos nodos nessa rede.

Embora existam muitos estudos nessa área, a contribuição desse trabalho está em transmitir mensagens de forma adaptativa, baseado na mudança de entropia de mobilidade do nodo. Através da entropia podemos medir o nível de atividade da rede o que trará resultados quanto à probabilidade de contato dos nodos. O que pode trazer ganhos quanto a economia de recursos nessas redes.

1.1 Objetivo

Gerenciar o armazenamento das mensagens nos *buffers*, combinando replicações de mensagens de acordo com a entropia de mobilidade a fim de evitar a sobrecarga da rede.

1.2 Objetivos específicos

Buscar uma relação entre a taxa de entropia de mobilidade e a probabilidade de contato dos nodos na rede, a fim de gerar um valor n correspondente às replicações da mensagem em um nodo específico. Com essa estratégia de repasse, será possível obter a administração das cópias de mensagens disseminadas na rede e controlar a sobrecarga gerada. Além de:

- Preservar os recursos de *buffer* disponível em cada nodo
- Não atingir *buffer overflow*
- Delimitar um ponto de alerta para a tomada de decisão

1.3 Organização do trabalho

As principais técnicas e mecanismos adotados para gerenciar *buffer* em DTN são vistas a seguir. No Capítulo 2 são apresentados os conceitos básicos necessários para o entendimento deste trabalho, é discutida também uma visão geral sobre alguns trabalhos relacionados encontrados na literatura que investigam técnicas de políticas de gerenciamento de *buffer*. O capítulo 3 apresenta a descrição do problema abordado neste trabalho. O capítulo 4 apresenta a metodologia adotada. Um resumo dos resultados encontrados é apresentado no capítulo 5. E finalmente, conclusão e trabalhos futuros estão no capítulo 6.

Capítulo 2

Fundamentação Teórica

Redes DTN têm o objetivo de prover conexões entre dispositivos em áreas que não são bem cobertas por serviço regular de rede. O conceito de redes tolerantes a atrasos e desconexões surge como uma solução para possibilitar a comunicação em cenários nos quais a conectividade entre os membros é intermitente e/ou existe grandes atrasos.

Alguns conceitos são necessários para que se compreenda o ambiente das Redes Tolerantes a Atrasos e Desconexões. A seguir, são apresentados os conceitos básicos necessários para o entendimento deste trabalho, será discutida também uma visão geral sobre alguns trabalhos relacionados encontrados na literatura que investigam técnicas de políticas de gerenciamento de *buffer*.

2.1 Conceitos Básicos

Nesta seção são apresentados conceitos sobre protocolos de roteamento, modelos de mobilidade e entropia de mobilidade e, alguns trabalhos relacionados ao uso de políticas de gerenciamento de *buffer* que são de fundamental importância para o entendimento deste trabalho.

2.1.1 Protocolos de Roteamento

A literatura mostra algumas soluções de técnicas associadas aos protocolos de roteamento que abordam o gerenciamento de *buffer*. O ideal seria não permitir o congestionamento do *buffer*, porém, em dispositivos móveis, por exemplo, onde a capacidade de armazenamento é muito limitada, exceder o limite é praticamente inevitável. Assim, diversas propostas foram estudadas e implementadas nos algoritmos de roteamento para controlar o revezamento do espaço disponível. A seguir, são descritas as principais características dos protocolos de roteamento mais reconhecidos para essas redes.

- *Direct Delivery*: este protocolo, diferente dos demais, não transfere mensagem para outro nodo, a menos que este seja o destinatário final da mensagem. Este pode ser considerado o algoritmo mais simples possível, pois um nodo A envia uma mensagem para um nodo B somente se esse for o destinatário da mensagem. Assim, esse esquema não possui limite para o atraso da entrega. A vantagem de sua utilização é que apenas uma única transmissão por mensagem é necessária.
- Epidêmico: Vahdat e Becker [3] propuseram um tipo de roteamento estocástico chamado Epidêmico, no qual o nodo origem difunde a mensagem para todos os seus vizinhos e cada um desses por sua vez a replica novamente. Este ciclo se repete até que a mensagem atinja o destino ou enquanto durar o tempo de vida da mensagem, quando este for especificado. Este processo de troca de mensagens se repete sempre que um nodo entra em contato com um novo vizinho. Dessa forma a mensagem é rapidamente distribuída em todos os nodos alcançáveis da rede. Então, quanto mais cópias de uma mesma mensagem forem encaminhadas, maior será a probabilidade de essa ser entregue e menor será o atraso. Algoritmos epidêmicos se mostram eficientes na entrega de mensagem, porem a um alto custo em termos do número de retransmissões (*overhead*) e consumo dos recursos dos nodos (*buffer*).
- *Spray and Wait*: Spyropoulos et al. [5] definem uma família de protocolos de múltiplas cópias. O protocolo *Spray and Wait* combina a velocidade do protocolo Epidêmico, com a simplicidade de envio direto para o destino do *Direct Delivery*. Seu objetivo é gerar um pequeno número de cópias a fim de assegurar que o número de transmissões seja pequeno e controlado. Nesse esquema, o nodo origem repassa todas as cópias para os primeiros L nodos distintos que ele encontrar. Se o nodo destino não for encontrado nessa fase, o protocolo entra na fase de espera, onde os L nodos que contem cópias da mensagem, irão repassa-la somente pra o nodo destino. É possível notar que o algoritmo depende de uma rede com mobilidade consideravelmente alta, pois em redes com poucos nodos móveis aumentam-se as chances de um destinatário permanecer isolado por muito tempo.

- *PRoPHET*: é um protocolo de roteamento probabilístico (*Probabilist Routing Protocol using History of Encounters and Transitivity*) proposto por Lindgren *et al.*[6]. Este estima uma métrica probabilística denominada “provisora de entrega” $P(A, B)$ sempre que um nodo A estabelece uma conexão com um nodo B. Esta métrica indica quais as chances que um determinado nodo (A) tem de entregar uma mensagem ao destino (B). As mensagens são repassadas somente para nodos com maior previsibilidade de entrega ao destino. O valor de $P(A, B)$ aumenta sempre que A e B se encontram e diminui se A e B deixam de se encontrar frequentemente. O tempo é controlado por uma constante K, denominada constante de envelhecimento, que corresponde ao número de unidades de tempo transcorridas desde a última vez que a métrica foi atualizada. Quando um nodo recebe a lista do seu vizinho, ele calcula a probabilidade de entrega para cada mensagem que ainda não possui. Em seguida, para cada mensagem, o nodo compara a probabilidade indicada na sua lista com a probabilidade indicada na lista recebida do vizinho. Essa comparação é realizada para verificar qual dos dois nodos possui maior probabilidade de entrega. Após essa comparação, três procedimentos são realizados: (I) o nodo envia um pedido das mensagens não armazenadas que possui uma maior probabilidade de ser entregue através dele; (II) recebe o pedido de mensagem do vizinho e as envia; e (III) apaga todas as mensagens que o vizinho tem maior probabilidade de entregar. No final, cada nodo possuirá somente mensagens cuja probabilidade de entrega seja maior através dele.

2.1.2 Transferência de Custódia

Esta transferência acontece quando um nodo precisa passar a responsabilidade da entrega de seu agregado para outro nodo intermediário de uma maneira confiável até que este chegue ao destino [7]. Um nodo não é obrigado a aceitar um pedido de custódia, no entanto, quando aceito só poderá descartar a mensagem se encaminhar o agregado a outro nodo que aceite a responsabilidade da custódia, ou ainda, quando o tempo de vida da mensagem expirar [1].

De acordo com a Figura 2, se uma mensagem M precisar ser enviada de uma origem s até um destino t que não está disponível no mesmo instante de tempo do envio, tal mensagem é custodiada a nodos intermediários que se encarregam de entregá-la ao destino.

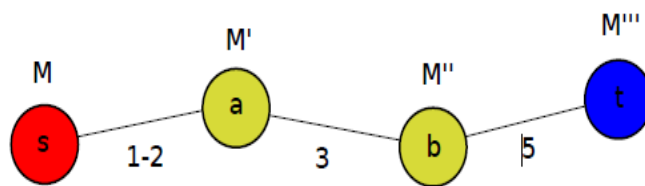


Figura 2 Custódias da mensagem M até entrega no instante 5.

Em DTN o acesso ao armazenamento é um dos recursos mais disputados, enquanto que em outras redes mensagens são simplesmente descartadas quando o *buffer* esgota sua capacidade, o mesmo não pode ser feito em DTN caso esteja com a custódia da mensagem. O ideal seria o armazenamento estar bem distribuído através da rede e os nodos possuírem capacidade de armazenamento suficientemente persistente e robusto para armazenarem agregados até o encaminhamento ocorrer.

2.1.3 Tempo de vida da mensagem (TTL)

Este mostra o tempo máximo que uma mensagem permanece ativa na rede, ou seja, o tempo a partir da criação da mensagem até que a última cópia seja descartada de um nodo [8]. Assim, mensagens permanecerão armazenadas em *buffers* enquanto durar o tempo de vida (TTL - *time to live*) dos dados a serem entregues. Dessa forma a rede pode descartar mensagens que podem ter perdido a importância a partir de um determinado tempo [9].

2.1.4 Número de replicações

Alguns protocolos de roteamento definem o número de cópias de uma mesma mensagem que pode ser disseminada pela rede. Diversas variações desse protocolo estão propostas em [5], todos com o objetivo de controlar o *flooding*, ou seja, o número de cópias da mesma mensagem na rede.

2.1.5 Notificação de recepção de mensagem

Nunes *et al.* [10], mostram outra técnica que utiliza mensagens de reconhecimento para informar aos outros nodos que a mensagem já foi entregue e então liberar espaço em *buffer* e melhorar a taxa de entrega. Cada vez que uma mensagem é entregue ao destino, um ACK é emitido através da rede notificando os nodos que ainda possuem uma cópia daquela mensagem do recebimento desta no destino. Um nodo, ao receber tal confirmação, remove a mensagem identificada de seu *buffer*.

2.1.6 Classes de prioridades

Permitir aos usuários que expressem uma relação de prioridade entre as mensagens da rede pode trazer um benefício considerável, tanto na forma como uma mensagem deva ser entregue antes de outra, quanto no momento de descarte, caso se tenha um *buffer* sobrecarregado [11].

A arquitetura DTN permite classificar os agregados em três níveis de prioridades: baixa (*bulk*), normal (*normal*) e expressa (*expedited*). Pode-se fazer uso dessa classificação e estabelecer prioridades quanto ao repasse das mensagens, transportando os agregados segundo seu grau de prioridade, sempre do maior para o menor [1].

2.2 Modelos de Mobilidade

Outra característica que influencia diretamente no desempenho da rede são os padrões de mobilidade dos nodos. Apesar de ser tratada como problema pelas frequentes desconexões e dificuldade na localização física dos nodos, pois seus endereços não estão associados a uma localização geográfica, a alta mobilidade também é utilizada como solução, conforme a Figura 3, confiando na mobilidade dos nodos para disseminar mensagens em direção ao destinatário [1].

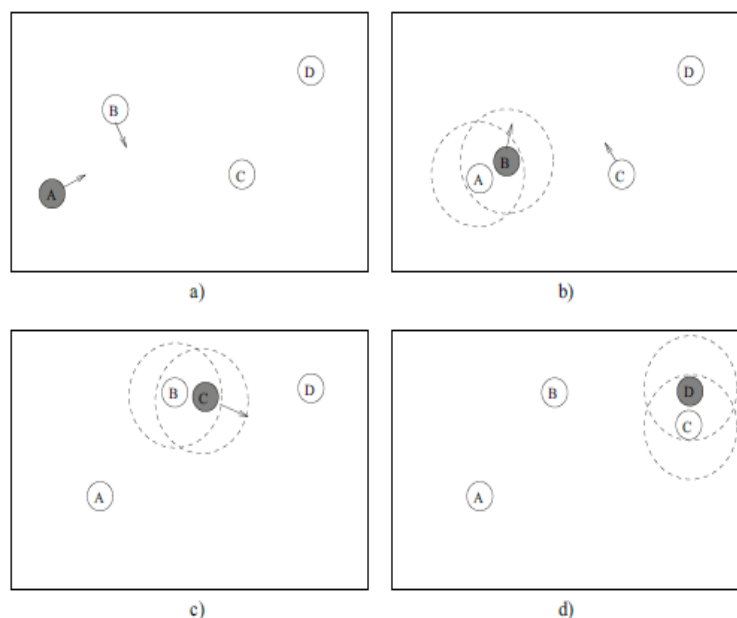


Figura 3 Mobilidade dos nodos para envio de mensagem ao destinatário.

Diversos modelos de mobilidade são utilizados nas análises de protocolos de roteamento em DTNs. No modelo *Random Way Point* [12] os nodos tem total liberdade para se movimentarem aleatoriamente em uma determinada área, no entanto não é um padrão muito realístico. Os nodos escolhem aleatoriamente um destino e se movem até ele.

Já nos padrões *Shortest Path Map Based Movement e Map Based Movement* [13], os nodos se movimentam levando em consideração informações de coordenadas que são importadas de um mapa de uma cidade, por exemplo. Utilizam o algoritmo de *Dijkstra* para encontrar o menor caminho até um destino selecionado de forma randômica, mas somente através dos caminhos disponíveis.

2.3 Entropia

Claude Shannon e Weaver [14] definem entropia como um conceito para medir o grau de incerteza que uma variável aleatória pode assumir, ou seja, representa a medida de desordem de um sistema. Um cenário com maior valor de entropia indica uma mobilidade mais dinâmica.

Silva [15] apresenta uma métrica de entropia do histórico de mobilidade que avalia o grau de previsibilidade de um nodo baseado em seu histórico de contatos.

Shannon [14] define que a entropia H de uma variável aleatória X dependente somente da distribuição de X , calculada pela Equação 1:

$$H(X) = - \sum_{i=1}^n p_i \log p_i \quad (1)$$

Assim, o conceito de entropia foi transformado por Shannon numa medida de quantidade de informação associada a uma distribuição de probabilidades.

An e Papavassiliou [16] utilizam entropia e propõem uma modelagem para suporte de rotas estáveis em redes *ad hoc* móveis. Através de informações de mobilidade (velocidade e direção) os protocolos de roteamento podem selecionar rotas estáveis entre origem e destino num ambiente onde múltiplos caminhos estão disponíveis, bem como uma medida de desempenho utilizada na avaliação da estabilidade e conectividade em redes *ad hoc* móveis.

Conforme descrito na referencia acima, o cálculo da entropia é feito localmente, cada nodo é responsável por conhecer sua situação em relação à rede. Então, considerando um

nodo móvel m e seu correspondente conjunto de nodos vizinhos chamados S_m , onde dois nodos são considerados vizinhos se um alcança o outro em um único salto (comunicação direta). Supondo que esses nodos trocam periodicamente informações de mobilidade, bem como informações locais (por exemplo, ID nodo) com seus vizinhos.

Denotando também que $v(m,t)$ representa o vetor velocidade do nodo m e $v(n,t)$ o vetor velocidade do nodo n , sendo considerados dois parâmetros velocidade e direção, a velocidade relativa $v(m,n,t)$ entre dois nodos m e n no tempo t é definida pela Equação 2:

$$v(m,n,t) = v(m,t) - v(n,t) \quad (2)$$

Em seguida, a mobilidade relativa entre qualquer par de nodos (m,n) durante um intervalo de tempo Δt é definida como a média da velocidade relativa absoluta ao longo do tempo Δt . Essas características variáveis representam uma medida de velocidade relativa absoluta entre dois pares de nodos. Por conseguinte, a variável considerada aqui é a mobilidade relativa, dada pela fórmula 3:

$$a_{m,n} = \frac{1}{N} \sum_{i=1}^N |v(m,n,t_i)| \quad (3)$$

onde N é o número de tempos discretos t_i que a informação de velocidade pode ser calculada e disseminada para os outros nodos vizinhos no intervalo de tempo Δt .

Com base nisso, encontra-se P_k como sendo a velocidade relativa de um nodo m para um dado nodo k dividido pelo somatório das velocidades relativas de todo o conjunto F_m no tempo Δt . Substituindo P_k na Equação 4, temos a Equação final da entropia $H_m(t,\Delta t)$ de um nodo móvel m para o conjunto dos nodos vizinhos ou qualquer subconjunto de S_m durante um intervalo de tempo Δt conforme Equação 4 descrita abaixo:

$$H_m(t,\Delta t) = \frac{-\sum_{k \in F_m} P_k(t,\Delta t) \log P_k(t,\Delta t)}{\log C(F_m)} \quad (4)$$

onde,
$$P_k(t,\Delta t) = \frac{a_{m,k}}{\sum_{i \in F_m} a_{m,i}}$$

Na Equação (4), F_m corresponde ao conjunto ou qualquer subconjunto que inclui todos os nodos vizinho do nodo m (exemplo: $F_m = S_m$) e $C(F_m)$ o grau de cardinalidade do conjunto F_m . E então a entropia é normalizada de modo que $0 \leq H_m(t, \Delta t) \leq 1$.

2.4 Trabalhos correlatos

Pesquisas na área de Redes Tolerantes a Atrasos e Desconexões apresentam uma significativa quantidade de trabalhos com soluções reativas, gerenciando *buffer* ou estabelecendo políticas de descarte de mensagens com o foco voltado para uma situação de estouro de *buffer*. Este trabalho traz uma proposta proativa, fazendo uso da previsibilidade de contato para definir um número máximo de cópias de uma mesma mensagem na rede, proporcionando um rodízio do espaço disponível para armazenamento, evitando a sobrecarga deste espaço e ficando sempre apto a novas demandas.

Dimitriou e Tsaoussidis [17] investigam um mecanismo que visa minimizar as transferências de mensagens entre *buffers* e uma forma de armazenamento persistente para acelerar as transmissões. Além de um *buffer*, cada nodo precisa ter uma unidade de armazenamento persistente, conforme mostra a Figura 4. O trabalho modelou um sistema em que os pacotes chegam de diferentes fluxos e investigou o potencial de manter esses dados entre *buffers* ou em armazenamento persistente para serem retransmitidos quando o próximo contato estiver disponível.

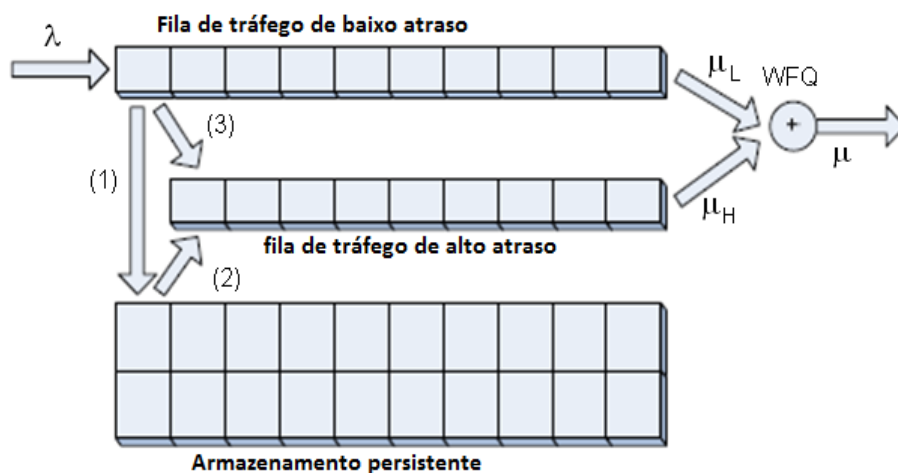


Figura 4 Modelo proposto em (Fonte: [17]).

Lindgren e Phanse [18], definem uma estratégia de gerenciamento de *buffer* que indica qual pacote será descartado quando o *buffer* de um nodo DTN está cheio e há um novo pacote para ser armazenado. Segundo os autores, a melhor opção é escolher a mensagem menos provável de ser entregue ao destino final. O ponto crítico é selecionar a mensagem a ser descartada, pois em uma topologia aleatória de uma DTN pode ser impossível prever isso. Seus experimentos avaliam quatro políticas de descarte, ou seja, para um cenário de *buffer overflow* qual a melhor estratégia de descarte, primeiro a recém-recebida, a mais antiga na fila, a com menor tempo de vida restante ou ainda a com maior TTL.

Seguindo a mesma ideia de como minimizar o impacto de *buffer overflow*, Kim *et al.* [19], apresentam um método que compara o número de cópias possíveis de uma mesma mensagem. Quando o estouro do *buffer* acontece, um nodo descarta a mensagem com maior número de cópias, o que resulta em uma maior taxa de entrega em uma DTN. A ideia por trás dessa proposta é proteger as mensagens com menor número de cópias para que essas possam ser duplicadas para mais nodos.

A fim de resolver este problema de como lidar com as mensagens no *buffer* de um nodo, Yun *et al.* [20], propõe uma estratégia de controle de congestionamento para o roteamento epidêmico em DTN, onde o descarte é baseado no número médio de repasses de cada mensagem (AFNER – *Average Forwarding Number based on Epidemic*), isto é, elegendo as mensagens que superam a média de encaminhamentos do *buffer*. Os resultados encontrados, após simulações em dois cenários distintos, foram comparados ao protocolo de roteamento epidêmico original e se mostraram superior em taxa de entrega, latência fim a fim e número médio de replicações.

Existe ainda uma proposta de classificação de mensagens segundo sua ordem de prioridade para redes veiculares tolerantes a atrasos e desconexões (VDTN) [21]. Conforme exemplificação na Figura 5, as mensagens na rede devem ser identificadas e classificadas de acordo com um esquema que define três classes de prioridades: baixa (*bulk*), normal (*normal*) e expressa (*expedited*). A estratégia propõe a criação de filas separadas para cada classe de prioridade, com recursos de *buffer* e tempo de utilização da rede dividida de forma proporcional a cada classe: expressa 60%, normal 30% e baixa 10%.

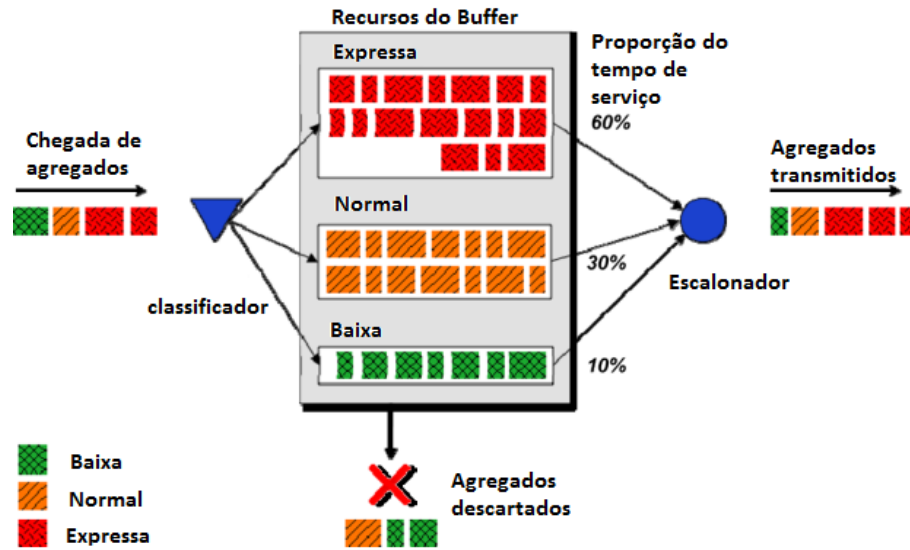


Figura 5 Buffer com classe de prioridade (Fonte: [21]).

Quando não houver espaço disponível para armazenar um novo pacote em uma fila de certa classe de prioridade, a política *First-in, First-out* (FIFO) descarta dessa fila os pacotes até que haja espaço suficiente disponível para armazenar a mensagem que chegou. Logo, todos os nodos da rede poderiam armazenar e transportar as mensagens de todas as classes de prioridade independente das restrições de armazenamento.

Outra abordagem, que servirá de base para este trabalho de pesquisa, está descrito na proposta de Ochiai e Esaki [22] que, embora voltada para roteamento permite alcançar altas taxas de entrega, transmitindo mensagens de forma adaptativa com base na mudança de entropia de mobilidade na rede. A estratégia chamada *Potential-based Entropy Adaptive Routing* (PEAR), que altera dinamicamente as replicações das mensagens na rede e são avaliadas de um ponto de vista de Ambiente Estruturado por Comunidade (CSE).

No entanto, o trabalho de Ochiai e Esaki não considera características importantes, como largura de banda e capacidade de armazenamento. Demonstram meramente a relação entre entropia de mobilidade e desempenho de roteamento.

A entropia tem sido amplamente utilizada para capturar o conteúdo de informação dentro de um sistema, seja para explorar os limites de previsibilidade na dinâmica humana para estudar os padrões de mobilidade dos usuários de telefones celulares anônimos [23], seja para modelar a estabilidade de rotas em redes móveis *Ad hoc*, utilizando o conceito de entropia e informações de mobilidade dos nodos, conforme descrito por Robinson *et al.* [24].

Capítulo 3

Descrição do Problema

Confiando-se na mobilidade da rede, políticas simples de roteamento surgem como solução para o problema da entrega de pacotes espalhando cópias de uma mensagem em diversos nodos.

Tais protocolos supõem que quanto maior o número cópias de uma mesma mensagem disseminada na rede, maior é a probabilidade de entrega da mesma, ou ainda, aumentam-se as chances de se encontrar o destinatário quanto maior for a proporção de nodos que carreguem a mensagem a ser entregue.

O roteamento epidêmico, algoritmo não probabilístico mais conhecido e utilizado para entrega de pacotes em redes móveis DTN, suporta a entrega das mensagens aos seus respectivos destinos sem o mínimo de conhecimento da topologia da rede, ou ainda sem que haja um caminho fim a fim entre origem e destino.

O mecanismo de transmissão das mensagens é semelhante a uma doença epidêmica, que contamina sua vizinhança com cópias de uma mensagem a ser transmitida para todo vizinho que entre em seu alcance de comunicação. A ideia é que em algum instante todos os nodos da rede estejam contaminados com a informação. Fazendo uso da mobilidade dos nodos para incrementar a probabilidade de entrega da mensagem a um destinatário específico.

O algoritmo de roteamento epidêmico possui uma alta confiabilidade ao supor uma vasta gama de recursos da rede para garantir o seu ideal funcionamento, tais como energia ilimitada e memória ilimitada, entre outros. No entanto, ao se pensar em redes móveis reais que impõem uma considerável limitação nos *buffers* dos nodos, percebe-se que a eficiência desta técnica é questionável. Devido à sua natureza, quase sempre baseado no espalhamento de mensagens, estes protocolos podem eventualmente gerar uma inundação na rede, ocasionando uma sobrecarga de recursos da rede.

O algoritmo *Spray-and-Wait* [5] surgiu com o objetivo de limitar a quantidade de cópias de uma mesma mensagem na rede, diminuindo a probabilidade de ocorrência de estouros de *buffer* e congestionamento. Quando a partir de certo limiar máximo de replicações de uma mensagem, começa a perder o desempenho ganho. O *Spray-and-Wait*, diferente do epidêmico, se utiliza do pressuposto de que L cópias de uma mensagem na rede é suficiente para adquirir um atraso esperado na entrega da mensagem, mantendo-se um nível aceitável de congestionamento dos recursos da rede. Trata-se de um esquema simples que produz um número limitado de cópias de uma mesma mensagem e proporciona uma boa taxa de entrega.

Devido ao grande número de troca de mensagem que pode ocorrer entre os nodos DTNs, a natureza desses algoritmos estimula o congestionamento da rede. O desafio é o tamanho dos recursos de *buffer* disponível e como esses nodos lidam com a possível escassez desse recurso. Uma estratégia já conhecida é organizar o espaço disponível em *buffer* de uma forma que seja conveniente para a rede visando algum critério específico.

Para isto surgiram algumas políticas que priorizam mensagens mais antigas, ou ainda priorizam mensagens mais recentes. Outras analisam parâmetros da própria mensagem, como seu TTL ou ainda se há algum critério de prioridade quanto ao envio da mensagem, e então decidem por aceitar ou rejeitar uma nova mensagem a ser transmitida.

O desafio é associar a eficiência dos protocolos de roteamento que inundam a rede produzindo cópias de uma mensagem a cada contato entre os nodos, com os recursos limitados dos dispositivos de comunicação.

Aperfeiçoar técnicas que gerenciem a utilização do espaço disponível para armazenamento em nodos DTN tem sido o foco de várias pesquisas em DTN. Muito se têm quanto a políticas de descarte que auxiliam na tomada de decisão em um cenário de estouro de *buffer*. No entanto, a principal motivação para a realização deste trabalho está em evitar uma situação reativa.

Neste trabalho, uma solução reativa é uma política que descarta as mensagens em caso de *buffer overflow*. Uma política bem simples de descarte em caso de *overflow* é o *Drop Random* [25]. Quando acionada, a política descarta as mensagens de forma aleatória, sem tirar proveito de certas características da rede e atribuindo um certo grau de incerteza no repasse de mensagens.

Outra política de gerenciamento reativo de *buffer* é o FIFO (*First In First Out*) ou

Drop Oldest [25]. Nesta política as mensagens são ordenadas e descartadas pelo TTL (*Time to Live*) dos dados armazenados. A ideia principal neste método é de que o pacote com o menor TTL no *buffer* tem uma maior probabilidade de já ter sido encaminhado e/ou entregue. No entanto, tais políticas estão sujeitas a descartar mensagens que ainda não foram difundidas pelos nodos da rede, visto que não utilizam informações sobre replicações das mensagens.

Lindgren e Phanse propõem a política MOFO (*Most Forwarded*) [18] visando aumentar a eficiência da disseminação das mensagens rede, esta política utiliza informações da rede para tomar decisão na hora do descarte. Então, descarta a mensagem encaminhada o maior número de vezes pelo nodo aumentando a probabilidade das mensagens menos encaminhadas serem difundidas, pois ficam mais tempo armazenadas em *buffer*.

Uma solução proativa seria utilizar a eficiência dos algoritmos não probabilísticos, ajustando o número máximo de disseminações de cópias de uma mesma mensagem na rede baseado no nível de perturbação da rede.

Fazendo-se uso do conceito de entropia que mede o grau de desordem de um sistema, a hipótese deste trabalho é condicionar o número de cópias que serão replicadas na rede ao nível de entropia de mobilidade que os nodos estão inseridos. A proposta é usar uma função inversamente proporcional entre entropia de mobilidade e número de replicações de uma mensagem. A fim de evitar uma eventual comparação errônea, evitou-se comparar essa política com algoritmos de roteamento, tendo em vista que, estes normalmente estão interessados no repasse de mensagens através da rede, enquanto que, esta proposta está preocupada com o nível de congestionamento dos recursos da rede, sem um objetivo específico no encaminhamento de mensagens.

Nesse sentido quando o nodo estiver em uma situação de muitos contatos, seja por ele ter muita mobilidade ou pela alta mobilidade de sua vizinhança, teremos um cenário com um alto grau de entropia. O que gera muita troca de mensagens e congestiona a rede e os *buffers*. Fazendo uso da proposta desse trabalho, quanto maior o grau de entropia menor será a quantidade de mensagens replicadas. Então cada nodo replicará as mensagens em seu *buffer* a quantidade de vezes indicada para aquele cenário em que se encontra na rede e removerá esta de seu *buffer*. Com isso abrirá espaço para novas possíveis mensagens nos próximos contatos e ainda garantirá que contribui para a disseminação desta na rede, pois o descarte obedeceu a um critério de replicações mínimo.

Numa situação contrária, quando o nodo estiver em um cenário de baixa entropia, este terá poucos contatos, logo não há a necessidade de ter muito espaço livre em *buffer*, pois a troca de mensagens acontecerá em menor quantidade e então as mensagens podem ser mantidas em armazenamento por mais tempo para serem replicadas mais vezes garantindo a disseminação dessas na rede.

Então, este trabalho provê uma técnica de gerenciamento de *buffer* em redes tolerantes a atrasos e desconexões baseada no controle do número de cópias da mensagem original de acordo com a entropia de mobilidade na rede. Supondo que a utilização da entropia indique o grau de concentração ou dispersão de movimento dos nodos é possível condicionar o número de replicações a entropia de mobilidade de modo a controlar e/ou evitar a sobrecarga da rede.

Capítulo 4

Metodologia

Inicialmente, foi realizado um estudo bibliográfico sobre entropia de mobilidade para planejar os experimentos. Para estes usou-se um ambiente virtual baseado no simulador *The ONE* [26], que será descrito posteriormente.

Os experimentos compreendem em variar o número de replicações das mensagens para um valor específico de entropia de mobilidade, a fim de relacionar as medidas de entropia, contatos e números de replicações, condicionadas uma à outra.

4.1 Simulador *The ONE*

O *Opportunistic Network Environment* (ONE) é uma ferramenta *open source*, desenvolvida em Java especificamente para simular redes DTN. Implementa os mecanismos necessários ao funcionamento de uma Rede Tolerante a Atrasos e Desconexões e ainda fornece uma ferramenta para simular situações específicas de movimentações de nodos em caminhos pré-determinados ou aleatórios, sendo possível personalizar a execução conforme seja necessário.

Seu funcionamento depende inicialmente da construção de um cenário de simulação por parte do usuário. Um cenário pode ser construído por meio da inserção de diversos elementos, como:

- Grupos de nodos
- Eventos
- Protocolos de roteamento
- Modelos de mobilidade
- Relatórios
- Mapas

Cada grupo de nodos possui características próprias, como velocidade de deslocamento, taxa de transmissão, tamanho do *buffer*, quantidade de nodos e identificador (ID). Diversos grupos podem ser inseridos no cenário possibilitando uma variedade de nodos presentes no ambiente de simulação.

Os ambientes a serem gerados na simulação nada mais são que mensagens (ou pacotes) com origem e destino determinados. Eles podem ser configurados em tamanho e intervalo de criação, além de poderem restringir quais nodos podem gerá-los e recebê-los.

Os protocolos de roteamento podem ser aplicados de maneira global ou local no ambiente de simulação, ou seja, cada grupo distinto de nodos pode usar seu próprio protocolo de roteamento. No ONE já estão implementados os protocolos Epidêmico, *Spray and Wait*, *MaxPro* e *PRoPHET*.

Os modelos de mobilidade são padrões de deslocamento de nodos e podem também ser aplicados de forma global ou local aos grupos de nodos. Dentre os diversos modelos, merecem destaque:

- *CarMovement*: modelo para carros, no qual o nodo possui um comportamento de seguir para o destino a partir de uma origem;
- *BusMovement*: modelo para ônibus, no qual o nodo possui um comportamento que alterna origem e destino continuamente (comportamento *ping-pong*);
- *ShortestPath*: modelo em que o nodo possui dois pontos de origem e destino e a menor rota entre tais pontos é calculado usando algoritmo de *Dijkstra*;
- *RandomWaypoint*: modelo em que são criados diversos caminhos aleatórios entre pares de pontos, com o nodo se deslocando por estes caminhos.

Os relatórios são os agrupamentos de dados gerados a partir da simulação. Eles fornecem estatísticas essenciais para a interpretação e análise dos diferentes cenários construídos, sendo os principais:

- Mensagens criadas: dados de mensagens que foram geradas no intervalo de simulação.
- Mensagens entregues: dados de mensagens que alcançaram com sucesso seu destinatário;

- Conectividade da rede: instantes de ocorrências de conexão e desconexão entre nodos.

Os mapas representam o conjunto de pontos e retas nos quais os nodos podem se deslocar. No simulador *ONE* é usado um formato específico de representação de mapas, o formato *WKT (Well Know Text)* [27]. A linguagem *WKT* permite de forma simples e direta descrever objetos geométricos como pontos, retas, polígonos e composições destes mesmos objetos. Os dados são tipicamente convertidos de mapas de dados do mundo real ou criados usando Sistemas de Informação Geográficos (*SIG*) de programas como *OpenJUMP*¹.

De fácil compreensão, a interface do *ONE* é composta por quatro áreas principais: área de simulação, controle da simulação, lista de nodos e *log* de eventos. Observam-se a interface gráfica e estas quatro áreas, na Figura 6;

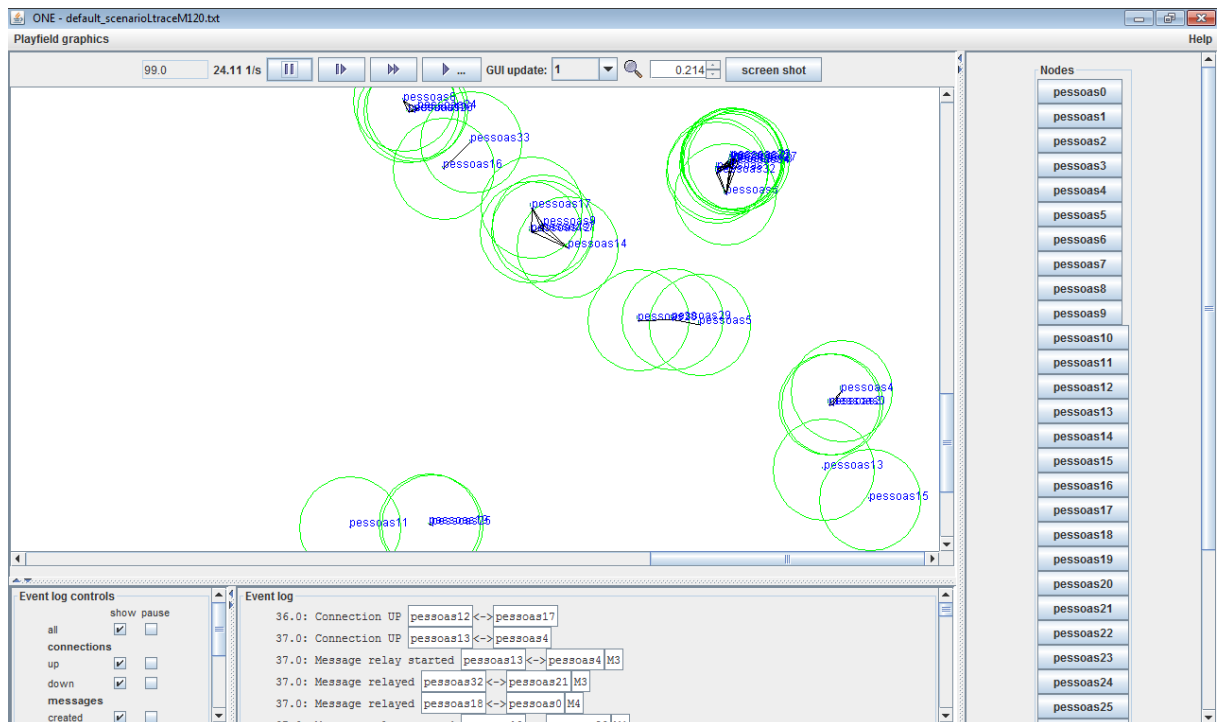


Figura 6 Interface gráfica do simulador *The ONE*.

4.2 Projeto de experimentos

O primeiro passo é a caracterização da entropia de mobilidade em redes DTN. Os fatores essenciais para a implementação do cálculo da equação (2) foram obtidos utilizando-se um ambiente virtual baseado no simulador *The ONE* [26].

¹ <http://openjump.org>

Entropia de mobilidade, segundo [16], está baseada na probabilidade da mobilidade relativa entre o nodo e o conjunto de nodos vizinhos a ele em um determinado intervalo de tempo Δt , em uma rede DTN, levando em consideração os seguintes fatores: velocidade e direção. Conforme mostra a equação (2), a entropia é normalizada resultando $0 \leq H_m(t, \Delta t) \leq 1$.

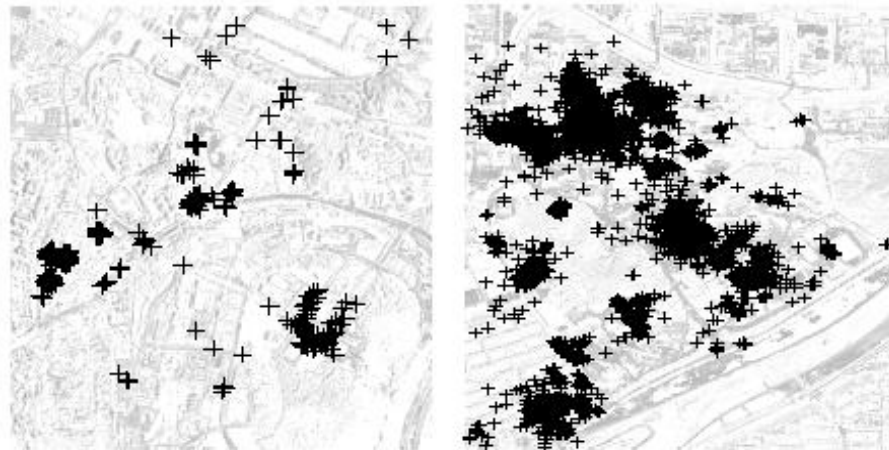
O gerenciamento de *buffer* reflete como as mensagens serão roteadas. Para este trabalho, usou-se o protocolo de roteamento epidêmico [3], um dos primeiros protocolos para DTN. Esse protocolo dissemina réplicas de uma mesma mensagem em todos os nodos alcançáveis na rede consumindo ao máximo os recursos disponíveis e resultando em baixa eficiência da rede em termos de taxa de entrega e sobrecarga de mensagens quando nodos têm tamanho de *buffer* e capacidade de transmissão limitada.

Para calcular a entropia, faz-se necessário selecionar modelos de mobilidade. Este trabalho utiliza modelos de mobilidade humana [28]. Os *traces* que são informações de usuários móveis em ambientes reais a partir de diferentes ângulos para caracterizar diferentes aspectos de mobilidade realista, coletados em dois locais diferentes que continham as coordenadas GPS necessárias ao cálculo, disponíveis no repositório CRAWDAD [29]. Tratam-se de coordenadas geométricas sobre um ponto de referência, para manter a privacidade dos usuários que fizeram as coletas. Um script foi usado para organizar e adequar o *trace* de coordenadas para deixar no formato do ONE.

Para avaliar essa proposta, simulamos dois conjuntos de dados diferentes com o protocolo epidêmico resultando em dois cenários distintos, mais informações dos *traces* e a disposição dos pontos visitados está descritas a seguir e na Figura 7 abaixo:

Traces NCSU: foram coletos em um campus universitário na Carolina do Norte por estudantes selecionados aleatoriamente do departamento de ciência da computação que carregavam também receptores GPS em suas atividades diárias.

Traces KAIST: são traces de receptores GPS de um campus universitário em Daejeon, Coreia do Sul, feito por alunos que viviam em dormitórios no campus universitário.



(a) NCSU

(b) KAIST

Figura 7 Disposição dos pontos visitados.

Com o intuito de transmitir uma maior confiabilidade à geração de mensagens para o ambiente simulado, as mensagens foram criadas usando uma distribuição Poisson para gerar mensagens em janelas de tempo com duração de 10 segundos, isto é, em uma janela de tempo compreendida por 10 segundos era escolhido um valor sob a visão Poisson determinar o tempo de criação da mensagem nesta janela de tempo.

Para medir a eficiência da proposta de gerenciamento de *buffer* e evitar descarte de mensagens pelo TTL, decidiu-se não atribuir valor ao tempo de vida das mensagens, portanto as mensagens nunca expiram nessas simulações. Parâmetros das simulações e características dos cenários estão resumidos na Tabela 1.

Tabela 1 Características das simulações

Parâmetros	NCSU	KAIST
Número de nodos	34	92
Raio de transmissão	200m	200m
Tamanho do <i>buffer</i>	0.5 - 2MB	0.5 - 2MB
Tempo da simulação	22h52min	22h37min
TTL	∞	∞

Os experimentos executam uma análise de sensibilidade, através da variação do número de replicações das mensagens para intervalos de valores de entropia de mobilidade, de acordo com a seguinte metodologia. Durante as simulações, utilizamos o protocolo epidêmico agregado a políticas reativas que tratam apenas do descarte em caso de *buffer*

overflow. Primeiro combinamos o epidêmico com a política FIFO (*First In, First Out*), pela qual uma situação de estouro de *buffer* está política usa da estratégia de remover da fila o primeiro elemento que tiver sido inserido, ou seja, não considera quaisquer informações sobre replicações das mensagens. Em outro momento, utilizamos a política MOFO (*Most Forwarded*) [18] que quando acionada descarta a mensagem que já foi mais encaminhada por aquele nodo.

Para a validação da proposta, os experimentos foram conduzidos como explicado a seguir. Na simulação, cada nodo acompanha a ocupação de seu *buffer* e, quando este chega ao nível de 70% de ocupação (escolhido como um estado de alerta), a simulação calcula a entropia para conhecer sua atual situação na rede e saber que decisão tomar naquele instante, ou seja, qual o número de vezes que as mensagens que ocupam seu *buffer* serão replicadas nos próximos contatos. Entropia é uma grandeza cujo valor varia de 0 a 1. Adotou-se neste trabalho intervalos de entropia de 0,1 unidade. Portanto, há dez níveis de entropia a serem associados ao número de replicações. Desse modo, quanto maior o grau de entropia, menor será o número de cópias que devem ser enviadas para a rede, pois elas mais facilmente irão ser replicadas pelos demais nodos.

Para encontrar um valor máximo de replicações de mensagens, utilizou-se um cenário com as características descritas na Tabela 2. O objetivo é simular uma rede medianamente esparsa com muita mobilidade, isto é, com uma grande variedade de nodos “perturbados” na rede, para obter maior discrepância entre os valores de entropia.

Tabela 2 Cenário avaliado para encontrar o tamanho de K

Número de nodos	500
Tamanho da área	7500x7500m
Área de cobertura	~35%
Entropia média da rede	~66,26%
Tamanho do <i>buffer</i>	2MB
Geração de mensagem	10s
Tempo de simulação	10000s
Modelo de mobilidade	Random Walk

Uma rede com maior valor médio de entropia não foi utilizado porque a geração de mensagens era alta.

Primeiro passo em busca de um limiar máximo para o número de replicações chamado aqui de K , foi utilizar diferentes valores de limite de replicações na rede. Comparando valores de sobrecarga e atraso médio. Variando de 2 em 2 um valor K de replicações, cujas médias são descritas na Tabela 3:

Tabela 3 Resultados das simulações

Tamanho máximo de K	Sobrecarga	Atraso médio(s)
6	1007,42	$3,514 \cdot 10^3$
8	1066,45	$3,409 \cdot 10^3$
10	974,17	$3,330 \cdot 10^3$
12	1218,96	$3,392 \cdot 10^3$
14	1233,56	$3,794 \cdot 10^3$

Percebeu-se que 10 era um limite superior favorável antes de se obter valores indesejados de sobrecarga, aumentando o atraso médio das mensagens, ou seja, representa a média com maior equilíbrio entre as métricas avaliadas.

A Tabela 2 enumera dois modos diferentes de selecionar o número de replicações de acordo com o grau de entropia, chamados de R1 e R2, a seguir detalhes das abordagens dos experimentos:

Tabela 4 Número de Replicações

Graus de Entropia	R1	R2
$0 \leq \text{Entropia} < 0,1$	10	10
$0.1 \leq \text{Entropia} < 0,2$	9	8
$0.2 \leq \text{Entropia} < 0,3$	8	8
$0.3 \leq \text{Entropia} < 0,4$	7	6
$0.4 \leq \text{Entropia} < 0,5$	6	6
$0.5 \leq \text{Entropia} < 0,6$	5	4
$0.6 \leq \text{Entropia} < 0,7$	4	4
$0.7 \leq \text{Entropia} < 0,8$	3	2
$0.8 \leq \text{Entropia} < 0,9$	2	2
$0.9 \leq \text{Entropia} \leq 1$	1	1

As escolhas dos valores de replicações para cada valor de entropia foram determinadas empiricamente após sucessivas rodadas de experimentos. O primeiro, chamado R1, foi

escolhido como referência pela variação linear reversa com a entropia. Após exaustivas rodadas de experimentos, percebeu-se que, à medida que os mesmos nodos permaneciam vizinhos por mais tempo a entropia tendia a crescer.

Então, para avaliar essa tendência utilizou-se também um campo limiar de 0.2 de entropia enquanto que para cada campo decrementado o seu número equivalente de replicações era diminuído em duas unidades chamado de R2, esperando diminuir mais a sobrecarga da rede e aumentar a sobrevida dos pacotes no *buffer*. A Figura 8 a seguir apresenta um gráfico mostrando a relação entre os valores de replicações abordados pela entropia:

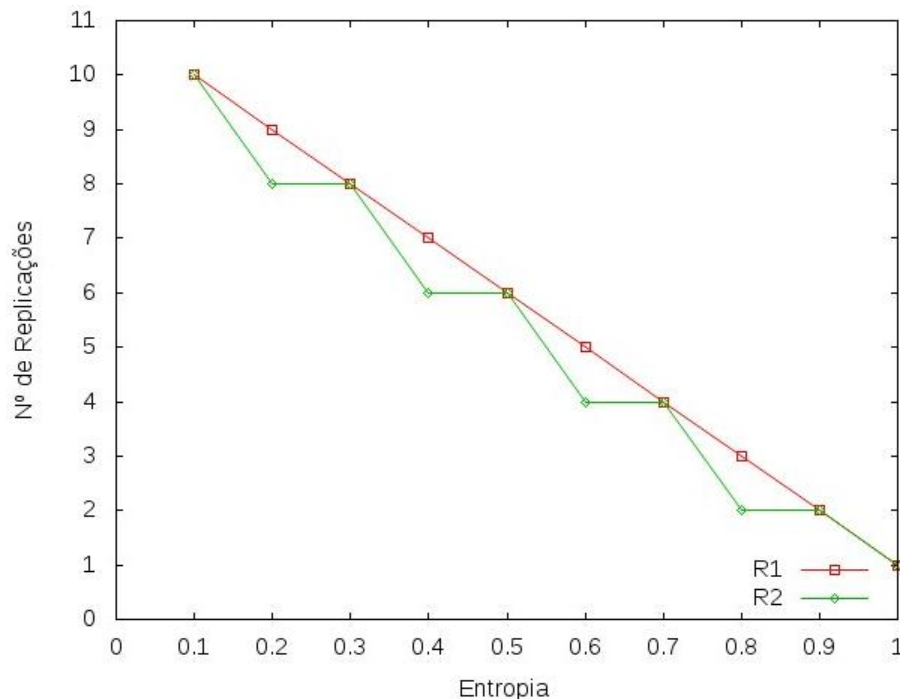


Figura 8 Relação entre replicações e entropia.

Por ser caracterizado como um trabalho preventivo para não estourar o *buffer* de um nodo DTN, a principal métrica de interesse é a taxa de ocupação dos *buffers*. Essa é a variável de resposta, uma vez que esse trabalho visa a não sobrecarga da rede e ter um valor ótimo de taxa de ocupação do *buffer* pode implicar em uma taxa menor de entrega ou ainda um atraso maior.

Por fim, os resultados obtidos são comparados com o caso em que o protocolo de roteamento epidêmico foi implementado com as políticas reativas de gerenciamento de *buffer* FIFO e MOFO.

Capítulo 5

Resultados

A política proposta neste trabalho foi avaliada através de simulação em dois cenários distintos de mobilidade. Cada cenário foi simulado com duas políticas reativas de gerenciamento de *buffer* e comparado com o caso em que se agregou essa proposta. As métricas utilizadas para avaliar o desempenho da proposta são:

- Taxa de ocupação dos *buffers*: é definida como a razão entre o tamanho total dos *buffers* ocupados e o tamanho total dos *buffers* existentes.
- Taxa de entrega de mensagem: é definida como mensagens criadas que chegaram ao destino durante a simulação. Dá-se pela razão entre mensagens entregue ao destino pelas mensagens criadas.
- Sobrecarga da rede: é definida como a quantidade de mensagens adicionais que foram transmitidas para cada mensagem que alcançou o destino durante a simulação. Dá-se pela razão entre a diferença do número de mensagens retransmitidas mt e mensagens entregues me e o número de mensagens entregue me .

Os resultados foram obtidos com nível de confiança de 95%, utilizando-se simulações de horizonte finito, repetidas 10 vezes cada simulação, tamanho da amostra necessária para alcançar dado nível precisão conforme fórmula para determinar tamanho das amostras descritas em Jain [30]. Os detalhes dos resultados estão apresentados por métricas nas subseções posteriores.

5.1 Taxa de ocupação dos *buffers*

Obteve-se para os dois cenários avaliados uma melhor ocupação média de *buffer* dos nodos em relação às políticas reativas usadas para comparação, FIFO e MOFO. Contudo, com uma margem de vantagem não superior a 10% em nenhum cenário avaliado, como por exemplo,

mostra a Figura 9 a maior diferença encontrada foi quando se avaliou os nodos do cenário de mobilidade NCSU com tamanho de *buffer* de 1,5MB, e a proporção de desempenho teve um ganho em torno de 7% apenas entre a política R1 e a política reativa FIFO.

No cenário NCSU, houve uma diferença de variação baixa para a média de ocupação dos *buffers*. Isso ocorreu porque se trata de um cenário mais esparsos, onde o grau de entropia esteve em um nível menor resultando em uma quantidade de replicações de mensagens maior pelo baixo nível de contatos entre nodos. Ou seja, as mensagens acabavam ocupando por mais tempo os *buffers* para ser replicada a quantidade de vezes correspondente ao seu grau de entropia.

No KAIST, tal como mostra a Figura 10, essa taxa de variação foi mais acentuada principalmente quando se utilizou 1,5MB de *buffer*/nodo. Pois sendo um cenário mais denso os nodos alcançaram altos valores de entropia devido ao elevado nível de perturbação na movimentação dos nodos resultando uma quantidade de replicações menor. Sendo assim, mais contatos aconteceram ao longo da simulação, logo as mensagens eram replicadas um número menor de vezes correspondendo ao seu elevado nível de entropia e mais rapidamente eram descartadas pelos nodos, liberando espaço em *buffer*.

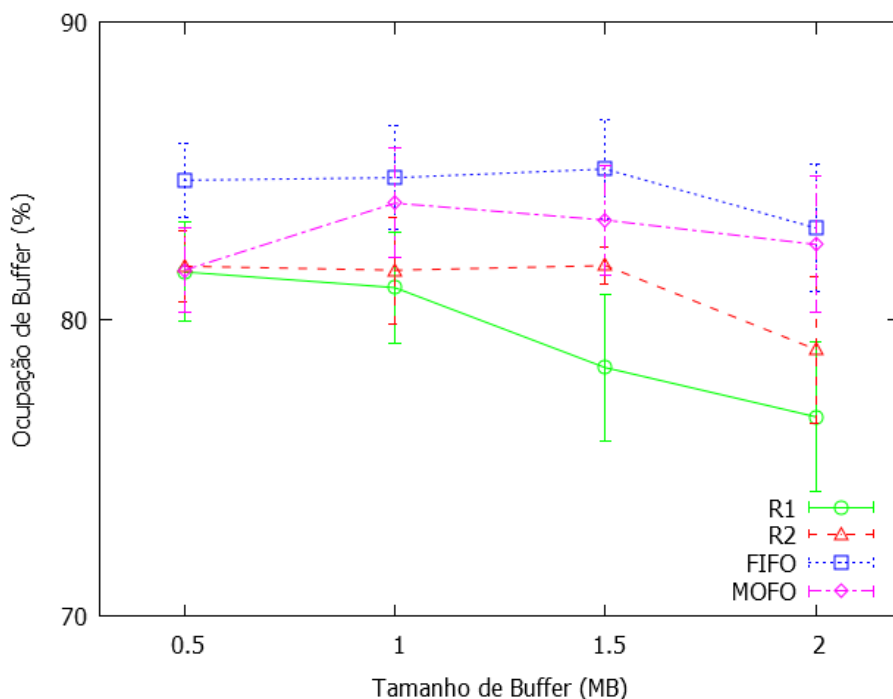


Figura 9 Cenário NCSU para ocupação de *Buffer*.

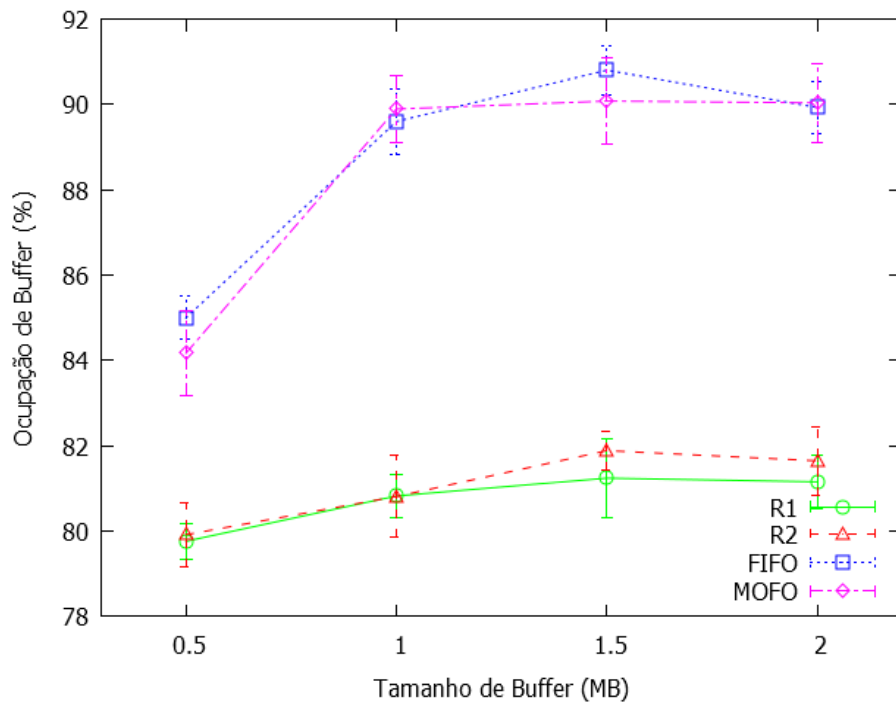


Figura 10 Cenário KAIST para ocupação de Buffer.

Contudo, percebe-se que o tempo que leva para o esgotamento dessa capacidade de armazenamento nos nodos, por mínimo que seja é favorável para que o tempo, pois permite que as mensagens permaneçam mais tempo armazenadas, o suficiente para aumentar a probabilidade melhora na taxa de entrega e também prover um atraso significativo na sobrecarga da rede, pois evita que mensagens sejam descartadas por falta de espaço de armazenamento dos nodos sem a garantia de terem sido disseminadas na rede.

As políticas reativas atingiram um grau de ocupação maior, como se previa. Percebe-se ainda que o método proposto na combinação do R1 tende a ocupar menos o *buffer* ao longo do tempo, pois na prática evita mais replicações de conteúdos na rede que o RII. Acredita-se que em um período muito amplo de observação de mobilidade RI possa se distanciar mais de RII em taxa de ocupação média de *buffer*.

5.2 Taxa de entrega de mensagens

As Figuras 11 e 12 mostram a taxa de entrega para os dois cenários de mobilidade NCSU e KAIST, respectivamente, nos quais os *traces* variam a quantidade de nodos e nível de mobilidade na rede. Para os dois cenários e todas as políticas, o comportamento é semelhante: quanto maior o tamanho do *buffer*, maior a taxa de entrega. Isso ocorre, pois com espaço de

armazenamento maior os nodos carregam mais mensagens e logo aumentam a probabilidade da troca de mensagens a cada contato. As políticas MOFO e FIFO possuem um desempenho similar sempre inferior ao desempenho do método proposto, reflexo da sobrecarga causada pelo número de cópias excessivas e que enchem os *buffers* e provocavam o descarte das mensagens. A Figura 11 apresenta os dados de taxa de entrega para o cenário NCSU. É interessante notar, que no cenário mais restrito de tamanho de *buffer*, a política MOFO obteve desempenho inferior dentre as políticas analisadas. Enquanto que o método proposta na forma de R1 e R2 obtém desempenho superior estatisticamente equivalente a taxa de entrega mesmo para *buffers* com até 2 MB de capacidade.

Isso se explica pelo fato da proposta estar limitando o número de cópias das mensagens replicadas e descartar com a garantia que contribuiu para a disseminação da mesma na rede, mostrando que pode haver um equilíbrio quanto ao número de replicações das mensagens na rede, nesse caso condicionado ao nível de entropia de mobilidade do nodo. Ao contrário das políticas que associadas ao protocolo epidêmico puro replica a cada contato promovendo sobrecarga e dependendo das políticas reativas de descarte sempre que o *buffer* encher, independente de já ter encaminhado a mensagens que descartou. O que torna a sobrevida da mensagem maior aumentando também a probabilidade de uma melhor taxa de entrega.

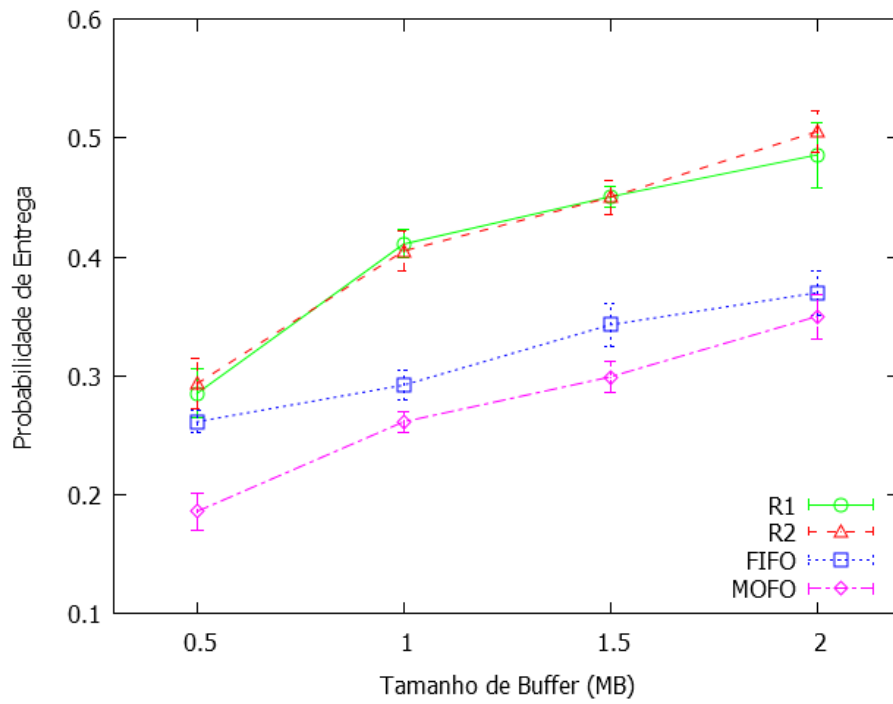


Figura 11 Cenário NCSU para probabilidade de entrega.

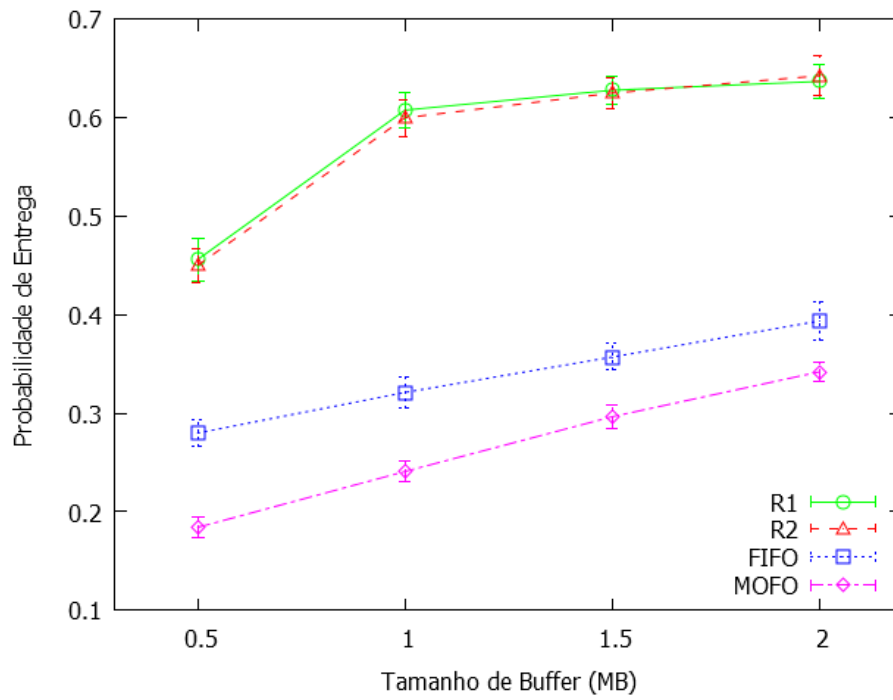


Figura 12 Cenário KAIST para probabilidade de entrega.

5.3 Sobrecarga da rede

Na avaliação dessa métrica, obteve-se a maior diferença entre as políticas reativas avaliadas em comparação as políticas propostas neste trabalho usando entropia de mobilidade para estabelecer um número de replicações de cada mensagem na rede. Nota-se que quanto maior o *buffer*, menor a sobrecarga para todas as políticas. Isso ocorre porque há mais espaço para armazenamento das mensagens, e reflete na melhor taxa de entrega também aumentada quando o tamanho do *buffer* cresce. O que também justifica o limite de *buffer* utilizado nesses experimentos, mostrando que o uso da política proativa aqui apresentada só é interessante quando os recursos disponíveis pelos nodos em questão de armazenamento são escassos.

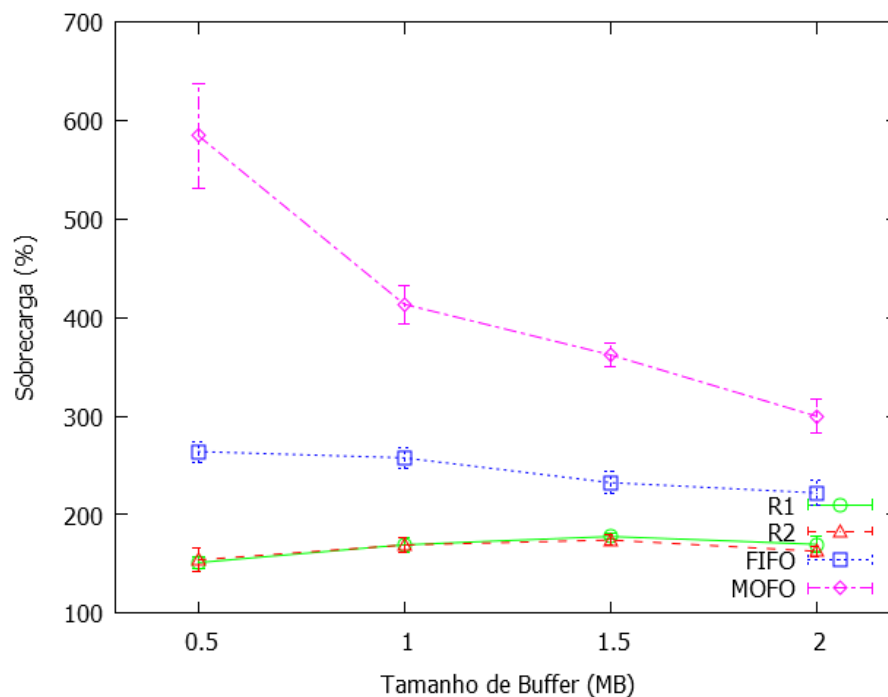


Figura 13 Cenário NCSU para sobrecarga.

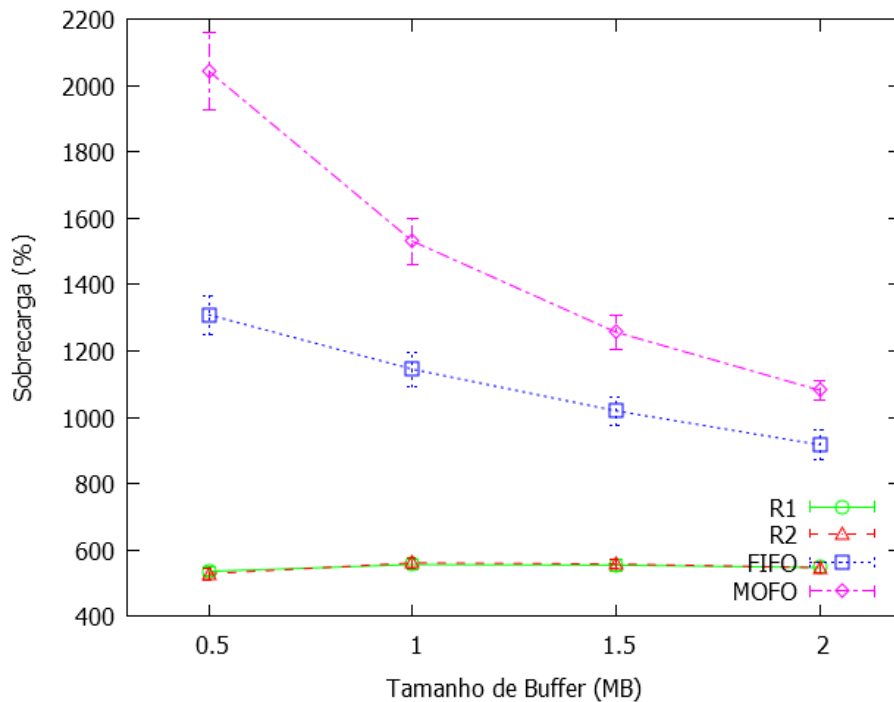


Figura 14 Cenário KAIST para sobrecarga.

A Figura 14, por exemplo, apresenta os resultados para o cenário KAIST que expressa o maior ganho que se teve nessa métrica. Mostra um cenário com maior número de nodos e mobilidade o que reflete num maior nível de entropia média na rede. Isso justifica a diferença entre o método proposto para as outras políticas comparadas, pois consegue se adaptar as condições da rede, logo maiores valores de entropia influenciam melhores resultados para a política proativa aqui apresentada.

Este resultado é justificado pela forma como a rede se comporta em relação a entropia de cada nodo individual. Se um nodo tem alguma estabilidade com os seus vizinhos, então não existe qualquer razão para replicar muitas vezes a mesma mensagens na rede uma vez que esses nodos já teriam trocado as mensagens entre si e então seus respectivos *buffers* estariam com as mensagens em comum. Ao mesmo tempo em que quando um nodo tem muita mobilidade e é instável quanto a sua vizinhança, mais contatos acontecerão e a troca de mensagens incomuns com seus vizinhos aumentará. Logo, a quantidade de replicações será maior. Dessa forma, pode-se limitar a quantidade de conteúdo gerado na rede usando apenas a estabilidade do nodo para sua vizinhança, evitando um grande número de replicações.

Capítulo 6

Conclusão e Trabalhos Futuros

As DTNs são um tipo de rede de grande importância por não necessitarem de requisitos de conectividade fim a fim, conseguindo aproveitar de maneira oportunista as características de mobilidade dos nodos para o seu funcionamento. Entretanto, são redes onde o problema da escassez de recursos pode afetar bastante o seu desempenho. Este trabalho apresentou um mecanismo de gerenciamento de *buffer* proativo baseada em entropia de mobilidade. Através de simulações se avaliou o desempenho dessa proposta em dois cenários distintos de mobilidade chamados NCSU e KAIST, e comparou com dois mecanismos existente na literatura as políticas de gerenciamento *buffer* reativa FIFO e MOFO.

Os resultados mostram que depois de avaliadas três métricas, o emprego do mecanismo proativo de gerenciamento de *buffer* fornecem ganhos significativos em relação aos cenários de comparação. O uso de replicações limitado pela entropia de mobilidade do nó mostra uma maneira eficiente de reduzir a quantidade de transmissões realizadas pelos nodos melhorando a taxa de entrega, diminuindo a sobrecarga da rede e preservando os recursos de *buffer* dos nodos melhorando o gerenciamento dos mesmos.

A taxa de entrega teve um ganho de até 63% no maior *trace*, o KAIST, em relação a boa taxa de entrega já apresentada pelo caso reativo de gerenciamento de *buffer*. O maior ganho foi na taxa de sobrecarga para o maior *trace* KAIST, o que mostra que com o tempo em uma rede mesmo densa o *overhead* chegou a cair pela metade em relação ao caso de comparação.

Avaliou-se ainda o impacto do mecanismo proativo de gerenciamento para diferentes tamanhos de *buffer*. As métricas mostraram que em sua maioria perde-se o ganho inicialmente obtido com os cenários onde há escassez de recursos. Ou seja, o gerenciamento pelo uso do

mecanismos aqui proposto só tem validade nos cenários avaliados quando se tem até 2MB de *buffer* disponível.

Pretende-se avaliar o impacto da utilização de diferentes valores para o parâmetro de TTL no desempenho deste método de gerenciamento de *buffer* proativo. Bem como avaliar a proposta agregada a outros protocolos de roteamento, como por exemplo, o Spray and Wait que estabelece uma relação de congestionamento com um número máximo de replicações fixo. Também continuar a avaliar o comportamento dinâmico da rede, novas curvas de entropia versus número de cópias replicadas de modo a determinar uma melhor relação entre entropia de mobilidade e o número de replicações. O desafio está em validar uma função que relacione o grau de entropia do nodo com a quantidade de replicações.

Referências Bibliográficas

- [1] Oliveira, C. T.; Moreira, M. D. D.; Rubinstein, M. G.; Costa, L. H. M. K.; e Duarte, O. C. M. B.(2007). “Redes Tolerantes a Atrasos e Desconexões”. Minicurso do 25º SBRC, pp. 203-256.
- [2] Ari Keränen, Jörg Ott. (2007). “Increasing Reality for DTN Protocol Simulations”. Tech. rep., Helsinki University of Technology, Networking Laboratory.
- [3] Vahdat, A., E Becker, D (2000). “Epidemic Routing for Partially-Connected Ad Hoc Networks””. Relatório técnico, Duke University.
- [4] Pelusi, L., Passarella, A., and Conti, M. (Nov 2006). “Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks”. IEEE Comm. Magazine.
- [5] Spyropoulos, T., Psounis, K., and Raghavendra, C. (2008). “Efficient routing in intermittently connected mobile networks: the multiple-copy case”. *To appear in ACM/IEEE Transactions on Networking*.
- [6] Lindgren, A., Doria, A. and Schelen, O. (2004). “Probabilistic Routing in intermittently connected networks. Mobile Computing and Communications Review”. in *Proc. SAPIR*.
- [7] Fall, K., Hong, W. e Madden, S. (2003). “Custody transfer for reliable delivery in delay tolerant networks”. Relatório técnico, Intel Research.
- [8] Nunes, Cristina, (2010). “Roteamento Utilizando Potencial de Entrega em Redes Tolerantes a Atrasos”. Universidade Católica do Rio Grande do Sul, Porto Alegre, Brasil. Tese doutorado.
- [9] Poersch, André M., (2010). “Localização de Redes de Emergência Tolerantes a Atrasos e Desconexões”. Universidade Federal de Minas Gerais, Belo Horizonte, Brasil. Dissertação de Mestrado.
- [10] Nunes, C. M., Link, E., and Dotti, F. L., (2009). “Evaluating the impact of an acknowledgment strategy for APRP”. In *5th International Latin American Networking Conference*, pages 77–86.
- [11] Oliveira, Thiago R., (2010). “Um Modelo de Gerenciamento de Segurança Adaptativo para Redes de Emergência”. Universidade Federal de Minas Gerais, Belo Horizonte, Brasil. Dissertação de Mestrado.

- [12] Camp, T.; Boleng, J. e Davies, V. (2002). “A survey of mobility models for ad hoc network research”. *Wireless Communications and Mobile Computing*.
- [13] Ekman, F.; Keränen, A.; Karvo, J.; Ott, J., (2008). “Working Day Movement Model”. *In: First ACM SIGMOBILE International Workshop on Mobility Models for Networking Research*.
- [14] C. E. Shannon and W Weaver, (1948). “A Mathematical Theory of Communication”.
- [15] Silva, Alexandre M., (2006). “Modelagem e Proposta para Predição de Mobilidade em Ambientes Fechados”. Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil. Dissertação de Mestrado.
- [16] An, B. and Papavassiliou, S. (2002). “An Entropy-Based Model for Supporting and Evaluating Route Stability in Mobile Ad hoc Wireless Networks”. *IEEE Comm. Letters*, vol. 6, no. 8, pp. 328-330.
- [17] Dimitriou, S. and Tsaoussidis, V. (October 2009). “Effective *Buffer* and Storage Management in DTN Nodes”. E-DTN 2009, St. Petersburg, Russia.
- [18] A. Lindgren and K. S. Phanse, (January 2006). “Evaluation of queuing policies and forwarding strategies for routing in intermittently connected networks,” *in Proc. of IEEE COMSWARE*.
- [19] Kim, D.; Park, H. and Yeom, I. (2008) “Minimizing the Impact of *Buffer* Overflow in DTN”, *In 3rd International Conference on Future Internet Technologies (CFI08)*, Seoul, Korea.
- [20] Yun, L.; Xinjian, C.; Qilie, L. and Xiaohu, Y. (2010). A Novel Congestion Control Strategy in Delay Tolerant Networks. *IEEE Second International Conference on Future Networks*.
- [21] Soares, Vasco N.G.J.; Farahmand, Farid and Rodrigues, Joel J.P.C. (2010). “Traffic differentiation support in vehicular delay-tolerant Networks”. *Telecommunication Systems. Springer*.
- [22] Ochiai, Hideya and Esaki, Hiroshi (2008). “Mobility Entropy and Message Routing in Community-Structured Delay Tolerant Networks”. *In Asian Internet Engeneering Conference (AINTEC)*.
- [23] Song, C.; Qu, Z.; Blumm, N. and Barabási, A. (2010). “Limits of Predictability in Human Mobility”. *Science*, 327, pp.1018–1021.

- [24] Robinson, K.; Turgut, D. and Chatterjee, M. (2006) “An Entropy-based Clustering in Mobile Ad hoc Networks”. , in: Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control (ICNSC). pp.1 –5.
- [25] Animesha Garpal, Shweta Jain and Dinesh Singh Baghel (June 2012). “An Analysis Optimal *Buffer* Management Policy to Improve QOS in DTN Routing Protocol”. *International Journal of Computer Applications*. pp. 7-11.
- [26] Keranen, A. (2008). “Opportunistic Network Environment Simulator”. Special Assignment report, Helsinki University of Technology, Department of Communications and Networking.
- [27] A. Keränen, T. Kärkkäinen, J. Ott. (2010). “Simulating Mobility and DTNs with the ONE”. *Journal of Communications*, vol.5, n.2, pp.92-105.
- [28] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee and Song Chong, (2008). “Demystifying Levy Walk Patterns in Human Walks”, Technical Report, CSC, NCSU.
- [29] CRAWDAD. Disponível em: <<http://crawdad.org/meta.php?name=ncsu/mobilitymodels>> Acesso em: 15 de set. 2012.
- [30] R. Jain, (1991). “The Art of Computer Systems Performance Analysis”. John Wiley and Sons, Inc..