

# Uma Análise de Mecanismos de Remoção de Mensagens para DTN em Cenários de Mobilidade Humana

Elenilson da N. Gomes<sup>1</sup>, Carlos Alberto V. Campos<sup>1</sup>,  
Sidney Lucena<sup>1</sup>, Aline C. Viana<sup>2</sup>

<sup>1</sup>Departamento de Informática Aplicada (DIA)  
Universidade Federal do Estado do Rio de Janeiro (UNIRIO) – Rio de Janeiro, Brasil

<sup>2</sup>Institut National de Recherche en Informatique et en Automatique (INRIA)  
Orsay, France

{elenilson.gomes, beto, sidney}@uniriotec.br, aline.viana@inria.fr

**Abstract.** *The dissemination of copies of a message in the network is one of the techniques used in DTN, in order to improve the delivery rate and to decrease the delay. However, many of these copies remain in the buffer of intermediate nodes, even after the message is delivered at the destination. In this work, we investigate the use of techniques for removing messages already delivered to the destination in two different realistic scenarios using three routing protocols, Epidemic, Spray and Wait, and BUBBLE Rap.*

**Resumo.** *A disseminação de cópias de uma mensagem na rede é uma das técnicas utilizadas em DTN, a fim de melhorar a taxa de entrega e diminuir o atraso. No entanto, muitas dessas cópias permanecem no buffer de nós intermediários, mesmo depois da mensagem ser entregue no destino. Neste trabalho, investigamos o uso de técnicas para a remoção de mensagens já entregues no destino em dois cenários realísticos diferentes, utilizando três protocolos de roteamento, Epidêmico, Spray and Wait e BUBBLE Rap.*

## 1. Introdução

Em alguns casos, a comunicação fim-a-fim, pode não estar acessível o tempo todo, ou talvez nunca chegue a ficar disponível, havendo a necessidade de usar uma nova proposta de arquitetura, chamada de Redes Tolerantes a Atrasos e Interrupções, (*Delay/Disruption Tolerant Networks*) - DTN [Fall 2003]. Nessas redes o armazenamento de mensagens é persistente, pois é usado o mecanismo *store-carry-and-forward*, no qual um nó armazena uma mensagem para posteriormente, num novo contato, encaminhá-la [A. Vahdat 2000]. Para aumentar a chances de entrega e reduzir o tempo que uma mensagem leva para chegar ao destino, muitos protocolos de roteamento são propostos, visando disseminar várias cópias de uma mensagem na rede. Infelizmente, muitas dessas cópias permanecem armazenadas nos nós intermediários após a entrega no destino. O uso de mecanismos de remoção de mensagens visa diminuir a taxa de ocupação no *buffer* dos nós, permitindo uma melhor troca de dados na rede.

Pesquisas realizadas em redes móveis, na maioria das vezes, são baseadas em modelos analíticos ou simulações sintéticas, que não refletem o movimento real dos usuários [Kim et al. 2006], ou podem induzir a contatos não realísticos, influenciando diretamente o resultado da avaliação de desempenho da rede móvel em questão [Rhee et al. 2011].

Como não foi encontrado na literatura trabalhos avaliando os mecanismos de remoção de mensagens em cenários com características realísticas de mobilidade, neste trabalho temos como contribuição a avaliação detalhada de quatro mecanismos de remoção de mensagens com três protocolos de roteamento, em dois cenários de mobilidade real. Dentre os mecanismos analisados, estão os modelos matemáticos, IMMUNE e IMMUNE-TX, além do ReMO [Gomes et al. 2012], que também é uma implementação do modelo matemático VACCINE [Z. J. Haas 2006] e o mecanismo TTL 50% que remove mensagens que ultrapassem 50% do tempo total de simulação. Como os mecanismos implementados independem do protocolo de roteamento, foram utilizados três protocolos distintos, Epidêmico, Spray and Wait e BUBBLE Rap. Os resultados obtidos confirmam que os protocolos apresentam um desempenho superior quando associados ao ReMO.

Este artigo está organizado da seguinte maneira. Na Seção 2, os trabalhos relacionados são apresentados e discutidos. Uma avaliação dos mecanismos é apresentada na Seção 3. Na Seção 4, são apresentados e discutidos os resultados obtidos. Por fim, a Seção 5 contém a conclusão deste trabalho.

## 2. Trabalhos relacionados

Em [Yuen and Schulzrinne 2010], mostra-se que para limitar o número de cópias de mensagens, as mesmas são eliminadas através do uso do TTL (*time-to-live*). Assim, o TTL foi definido com base no tempo e com base no número de saltos na rede. Para o TTL baseado no tempo, inicialmente, o nó tem um valor definido para  $TTL_{max}$ . Este valor é decrementado a cada segundo até chegar a zero, quando a mensagem é apagada. As mensagens são replicadas, com o mesmo valor de TTL, quando o tempo expira, todas as mensagens são eliminadas simultaneamente. Este mecanismo será utilizado neste artigo, com o valor de 50% do tempo total de simulação.

Em [S. Kaveevivitchai 2010], é apresentado um mecanismo para eliminação de cópias de mensagens. O projeto foi separado em duas partes: (i) Distribuição de mensagens de confirmação (ACK) e (ii) Utilização de nós auxiliares para retransmitir mensagens de ACK. Os ACKs também foram divididos, ACKs ativos e ACKs passivos. O ACK passivo não vai ser encaminhado para qualquer nó, a menos que, receba uma cópia de mensagem já entregue no destino. As mensagens de ACKs passivos são distribuídas lentamente e as cópias são eliminadas em conformidade. Nos ACKs ativos, um nó que tenha ACK de qualquer mensagem tenta expressar seu conhecimento para qualquer nó que faça contato, (*broadcast-like*). O conceito de ACK passivo é igual ao conceito do IMMUNE, e o de ACKs ativos é equivalente ao VACCINE descritos em [Z. J. Haas 2006].

Em [Gomes et al. 2012] é apresentado o ReMO (Remoção de Mensagens Obsoletas), que tem como objetivo remover cópias de mensagens que já foram entregues ao destino e que permanecem ocupando espaço no *buffer* dos nós intermediários. Este mecanismo utiliza uma lista de mensagens entregues para cada nó. Assim, quando um nó entra em contato com outro, eles trocam suas listas e as atualizam. Depois, verificam em seus *buffers* se existe alguma mensagem que está nesta lista e caso exista, irão removê-la. Após esta fase, irão trocar as mensagens conforme o protocolo de roteamento empregado e, finalmente, se existir alguma mensagem que o nó destino seja ele, então, o identificador desta mensagem será armazenado na lista de mensagens entregues de ambos os nós. A grande vantagem do ReMO é que o mesmo independe do protocolo de roteamento usado. Entretanto, no trabalho [Gomes et al. 2012], a avaliação do ReMO não foi comparada

com outros mecanismos de remoção e não foram usados traços reais de mobilidade.

Em [Rhee et al. 2011], é realizado um estudo sobre os padrões de mobilidade humana e sua utilização em simulações de redes móveis. Verificaram que o padrão de mobilidade humana sofre influências externas, não sendo aleatório como se achava, pois a mobilidade está relacionada a contextos sociais. Além disso, observaram a existência de diferentes padrões de mobilidade relacionada a estes contextos. Por fim, consideram a importância do uso de modelos que sejam capazes de fornecer resultados mais realísticos nas simulações, como o uso de traços reais para aumentar a veracidade das simulações.

A preocupação em buscar cada vez mais realismo em simulações, também é o foco em [Thakur et al. 2011] no qual, é realizado um estudo sobre o uso de traços de mobilidade real em ambientes simulados. Para esta análise foram utilizados cinco traços de mobilidade real, que foram analisados com o protocolo Epidêmico e comparados com modelos de mobilidade. Os resultados obtidos mostram que os modelos de mobilidade ficaram distantes dos resultados obtidos através de traços reais. Assim, quando possível recomenda-se o uso de traços reais nas avaliações a serem realizadas.

Com base nos trabalhos acima descritos, na próxima seção, o presente artigo apresenta uma avaliação de diversos mecanismos de remoção para se verificar qual é o mais eficiente em dois diferentes cenários usando traços reais de mobilidade.

### **3. Avaliação de desempenho dos protocolos de roteamento sobre a influência dos mecanismos de remoção**

Espera-se que, com o uso de mecanismo de remoção, seja possível utilizar melhor o *buffer* dos nós, o que permitirá um melhor desempenho da rede e dos protocolos de roteamento que usam a disseminação de mensagens como forma de aumentar a taxa de entrega e diminuir o atraso. Para a implementação e avaliação dos mecanismos, foi utilizado o simulador de DTN (Opportunistic Network Environment - ONE).

#### **3.1. Descrição dos protocolos de roteamento utilizados**

Segundo [Moreira et al. 2012a], os protocolos roteamento/encaminhamento são classificados como: (i) baseados em inundação, (ii) baseados em replicação e (iii) baseados em encaminhamento de mensagens. Como os protocolos baseados em encaminhamento não geram cópias de uma mensagem, somente serão analisadas as duas classes anteriores. Entretanto, como existem muitos protocolos em cada classe, foram escolhidos: Epidêmico que é baseado em inundação, *Spray and Wait* que é baseado em replicação controlada e *BUBBLE Rap* que é baseado em replicação em função do contexto social.

Em [A. Vahdat 2000] é proposto o protocolo Epidêmico e seu funcionamento se dá da seguinte forma, cada nó tem uma lista de mensagens que estão no seu *buffer*. Ao entrar em contato com outro nó, os dois nós trocam suas listas e caso existam mensagens diferentes em seus *buffers*, irão trocá-las. Pode-se observar dois fatores importantes neste protocolo, o primeiro que como ele aumenta consideravelmente o número de cópias na rede, a probabilidade da mensagem chegar ao destino é maior. O segundo fator é que como ele dissemina várias cópias pela rede, provocando o enchimento do *buffer* dos nós, mais rapidamente. Esses fatores devem ser mitigados com mecanismos de remoção de mensagens e serão investigados no presente artigo.

Em [T. Spyropoulos 2005] é apresentado o protocolo *Spray and Wait*, no qual o processo de encaminhamento é dividido em duas partes: (i) a fase de pulverização (*spray*), onde  $L - 1$  cópias de uma mensagem são disseminadas na rede, e (ii) na fase de espera (*wait*), quando o nó repassa somente para o destino usando o mecanismo de “transmissão direta”. Para as simulações na Seção 4, o parâmetro  $L$  foi configurado com o valor 6.

O protocolo *BUBBLE Rap* [Hui et al. 2011] utiliza as relações sociais dos nós como decisão no encaminhamento de mensagens. O seu funcionamento é baseado nas métricas de centralidade e comunidade, onde cada nó participa de uma comunidade e sua centralidade é proporcional a sua popularidade (grau de conectividade do nó). Além disso, também tem uma centralidade global na rede para o encaminhamento de mensagens fora de uma comunidade. Possui duas fases, *Bubble-up* na comunidade global e *Bubble-up* na comunidade local, sempre escolhendo nós com mais centralidade para encaminhamento de mensagens. Para as simulações na Seção 4, o parâmetro  $K$  foi configurado com o valor 5 e o parâmetro `familiarThreshold` com o valor 700, conforme [Moreira et al. 2012b].

### 3.2. Mecanismos a serem comparados

Os mecanismos de remoção visam informar aos nós intermediários que uma mensagem foi entregue ao nó de destino, possibilitando a remoção de suas réplicas da rede. São também conhecidos como *Antipacket*, impedem que o nó receba novamente a mensagem já descartada. Outro mecanismo avaliado foi o TTL (Time To Live). Os mecanismos foram implementados conforme os modelos matemáticos citados em [Z. J. Haas 2006], podendo ser empregados em qualquer protocolo de roteamento. O mecanismo Immune tem a função de tornar imune o nó de uma mensagem obsoleta, isto significa que ao receber uma mensagem que já tenha sido entregue no destino este nó não aceite mais uma cópia desta mensagem, evitando inclusive as retransmissões desnecessárias. O mecanismo Immune-TX também tem a função de tornar imune o nó de uma mensagem obsoleta, mas ele vai além, imunizando o nó que tentou passar a mensagem obsoleta. Já o ReMO funciona como o VACCINE, imunizando todos os nós que tiverem contato conforme descrito em [Gomes et al. 2012].

### 3.3. Descrição dos parâmetros da simulação

Para a avaliação de diferentes cenários de uso de uma DTN, foram escolhidos dois traços de mobilidade real: UCL1 [Abdesslem et al. ] e RollerNet [Leguay and Benbadis ], disponibilizados no CRAWDAD <sup>1</sup>, que possuem diferenças significativas na mobilidade dos usuários. Os traços de mobilidade UCL1 foram obtidos por meio da movimentação de pessoas ao longo do campus da University College London. Já os traços do Rollernet são referentes ao movimento de patinadores em ruas da cidade de Paris. Nesses traços existe uma peculiaridade, que é o efeito sanfona no deslocamento dos usuários ao longo do tempo, devido a dois padrões de mobilidade existentes: um quando os patinadores se aglomeram, aguardando a liberação de algum cruzamento, e o outro quando estão patinando normalmente ao longo de uma via. Essa peculiaridade, gera uma alta conectividade entre os usuários periodicamente que pode influenciar o funcionamento da DTN diferentemente de outros cenários como o UCL1.

Para a geração de mensagens de ambos os cenários, UCL1 e Rollernet, foram criados 10 arquivos distintos de mensagens. No UCL1 cada arquivo de mensagem gerado

<sup>1</sup>CRAWDAD é um projeto que disponibiliza dados sobre experimentos reais de redes sem fio.

contém 8640 mensagens distribuídas aleatoriamente no tempo, entre os 20 nós ao longo dos 6 dias de simulação. No Rollernet cada arquivo de mensagem gerado contém 2160 mensagens distribuídas aleatoriamente no tempo, entre os 62 nós ao longo das 3 horas de simulação. Além disso, foi aplicado aos resultados obtidos o intervalo de confiança de 95% usando a distribuição *t-student*.

### 3.4. Métricas de desempenho utilizadas

Neste trabalho foram escolhidas as métricas probabilidade de entrega, atraso médio e sobrecarga de mensagens. Probabilidade de entrega de mensagens: número de mensagens entregues dividido pelo número de mensagens criadas durante uma simulação. Atraso médio: tempo médio que uma mensagem leva para ser entregue, desde quando é gerada até chegar ao seu destino. Sobrecarga de mensagens: quantidade de mensagens repassadas dividido pela quantidade de mensagens entregues durante uma simulação.

## 4. Resultados obtidos

Nesta seção os resultados obtidos são apresentados e analisados por métrica de desempenho. Primeiramente, a probabilidade de entrega é analisada, depois o atraso médio e a sobrecarga de mensagens.

### 4.1. Probabilidade de entrega

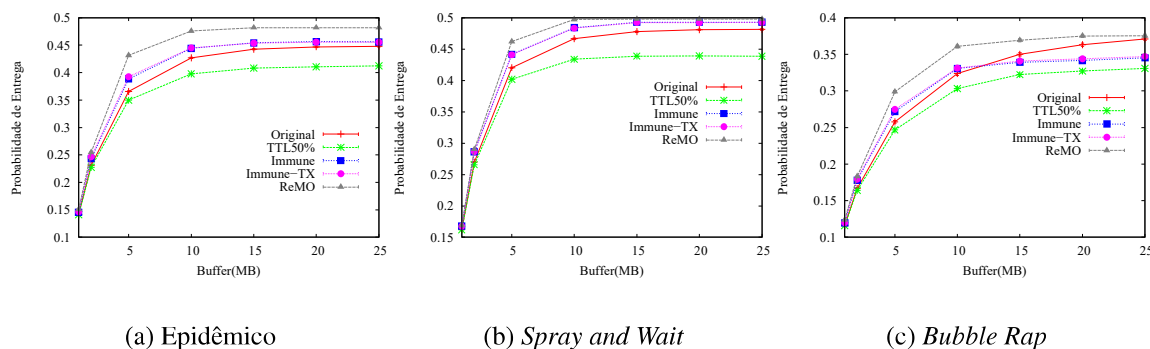
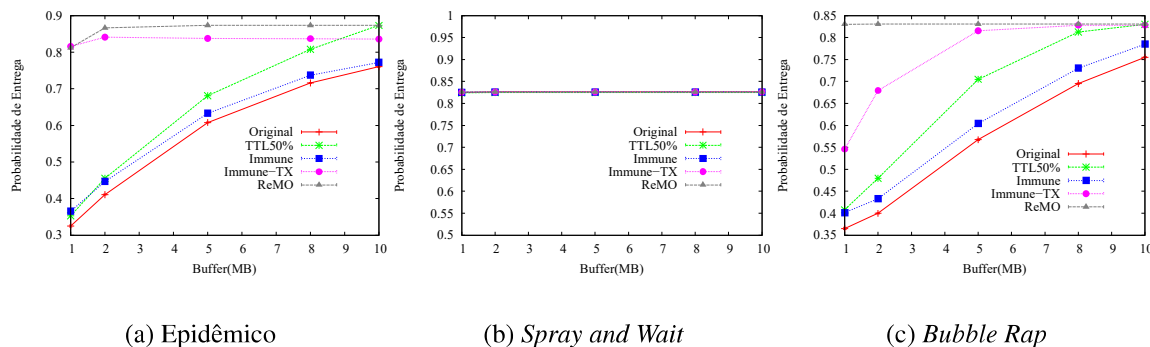


Figura 1. Probabilidade de entrega no cenário UCL1.

Na Figura 1 é analisada a probabilidade de entrega no cenário UCL1. Na Figura 1(a) para o Epidêmico, os mecanismos apresentaram uma melhora com a remoção de mensagens obsoletas, o descarte foi menor, aumentando a chance da entrega no destino. O mecanismo ReMO obteve um melhor desempenho também para este protocolo, isto se deve à remoção de mais mensagens obsoletas. Com isso, o *buffer* teve mais espaço para armazenar outras mensagens aumentando esta probabilidade. Como os outros mecanismos não removem tantas mensagens como o ReMO, o desempenho ficou abaixo, mas ficou um pouco melhor que o sem mecanismo. Já o TTL de 50% entre os mecanismos analisados foi o que obteve o pior resultado, isto deve-se ao fato, de que muitas mensagens foram criadas no intervalo de tempo inicial da simulação e não tiveram o tempo suficiente para chegar ao destino, sendo descartadas. Na Figura 1(b) para o *Spray and Wait*, observa-se que em sua implementação existe um controle na disseminação das mensagens na rede, com isso, o número de cópias é controlado. O ReMO apresentou um melhor resultado devido a sua implementação, que aumenta a remoção das mensagens obsoletas nos *buffers*. Já com o uso do TTL em 50% não obteve um bom resultado, pois, como já mencionando anteriormente, o descarte das mensagens no meio da simulação impediram a entrega de mensagens, influenciando diretamente no resultado. Na Figura 1(c) para o

*Bubble Rap*, é possível observar que para o *buffer* pequeno, entre 1M e 10M os mecanismos apresentaram um desempenho melhor e à medida que o *buffer* vai aumentando este desempenho, vai se igualando ao sem mecanismo e até mesmo piorando. Para esse traço de mobilidade com poucos nós, observamos que os mecanismos Immune e Immune-TX pioram o desempenho a partir de 12M, nesta fase o *buffer* não transborda mais, como estes mecanismos não disseminam a lista de mensagens entregues para todos os nós, trocam um número maior de mensagens obsoletas, influenciando de forma negativa nos resultados. Já o ReMO mantém um melhor desempenho até o 25M. O resultado para o TTL de 50% é o pior, pois como ele descarta as mensagens com o tempo médio de simulação, as chances reduzem bastante, visto também que o protocolo *Bubble Rap* utiliza o contexto social para encaminhar as mensagens até o destino, selecionando os nós com mais popularidade, como o tempo de vida das mensagens reduzem influenciando diretamente neste mecanismos piorando o seu desempenho.



**Figura 2. Probabilidade de entrega no cenário Rollernet**

A Figura 2 é analisada a probabilidade de entrega no cenário Rollernet. Na Figura 2(a) para o Epidêmico, observa-se uma melhora na métrica probabilidade de mensagens entregues com o uso do mecanismo ReMO, pois mantém o *buffer* abaixo de 80% da sua ocupação, mesmo quando o tamanho do mesmo é de apenas 1MB, favorecendo o recebimento de novas mensagens, por conseguinte aumentando a probabilidade de entregar mais mensagens ao destino. Esta melhora também ocorre com o Immune-TX, devido à movimentação dos nós, aumenta a funcionalidade deste mecanismo, chegando bem próximo ao funcionamento do ReMO. Já o Immune que não avisa aos nós intermediários mantém uma média melhor que o sem mecanismo, mas inferior até mesmo ao TTL de 50%, tem uma probabilidade melhor por ser favorecido pelo traço de mobilidade real. Na Figura 2(b) para o *Spray and Wait*, observa-se que para este cenário e com o uso deste protocolo que tem como característica a baixa taxa de ocupação do *buffer*, o uso dos mecanismos de remoção se mostraram pouco efetivos. É possível também observar que devido ao traço de mobilidade o uso do TTL de 50% também teve uma probabilidade de entrega igual. Na Figura 2(c) para o *Bubble Rap*, observa-se que a probabilidade de entrega para este protocolo com o mecanismo de remoção ReMO mantém-se estável numa probabilidade de entrega de aproximadamente em 83%, isto deve-se ao fato do mecanismo remover as mensagens obsoletas, aumentando o número de novas mensagens recebidas. O mecanismo Immune-TX também obteve um bom resultado, não superior ao ReMO, mas como também removeu mais mensagens que o outro mecanismo, obteve um resultado melhor. Já o TTL de 50% surpreende pelo seu desempenho, mas ao observarmos os contatos existentes neste traço de mobilidade real, verifica-se que na metade da simulação foram feitos 1851 contatos entre os nós, e na segunda metade foram feitos mais

3080, logo a probabilidade de mensagens serem entregues na segunda parte da simulação aumenta, explicando o resultado.

#### 4.2. Atraso médio

Na Figura 3 é analisado o atraso médio no cenário UCL1. Ao analisarmos os resultados apresentados na Figura 3(a), observa-se que o mecanismo ReMO apresenta o pior resultado. Isto ocorre porque o ReMO, ao remover mensagens que já foram entregues no destino, aumenta o espaço no *buffer* evitando que a política de gerenciamento de *buffer* funcione. Como a política configurada é a FIFO, que remove a mensagem mais antiga, se ela não atua, mensagens com mais tempo no *buffer* permanecem, aumentando à chance de chegar ao destino. Desta forma, a métrica atraso médio, que é definida como, tempo médio que uma mensagem leva para chegar ao destino, termina apresentando um resultado ruim. Para os outros dois mecanismos ocorreu a mesma situação, como removem um número menor, o atraso é menor em relação ao ReMO. O TTL de 50% obteve um resultado melhor em relação a todos os outros analisados, isto também se deve a definição desta métrica, pois como o tempo foi dividido em duas partes, logo as mensagens que chegaram antes desta metade obtiveram um tempo de criação e entrega pequeno, na segunda metade o tempo também será pequeno entre criação e entrega resultando neste resultado melhor. Na Figura 3(b) para o *Spray and Wait*, observa-se que os mecanismos tiveram um resultado não muito significativo, isto se deve a própria definição do protocolo de roteamento. Como este protocolo não dissemina muitas mensagens na rede, faz com que os resultados fiquem bem próximos tanto dos que utilizaram mecanismo como o sem mecanismo. Já para o caso do TTL de 50%, também ocorre a mesma coisa já analisando anteriormente, onde o tempo é dividido em duas partes, fazendo com que o tempo de criação e entrega sejam bem reduzidos. Na Figura 3(c) para o *Bubble Rap*, observa-se que o desempenho dos mecanismos são aparentemente piores, principalmente o ReMO, isto se deve à própria definição desta métrica, que contabiliza o tempo médio que uma mensagem leva para ser entregue, desde quando é gerada até o seu destino. Como os mecanismos descartam as mensagens já entregues no destino e ficam mais tempo com mensagens mais antigas que normalmente são descartadas quando o *buffer* está cheio, por este motivo até mesmo o desempenho de TTL de 50% foi melhor, pois como entregou mensagens novas, onde o tempo de criação e entrega eram menores, obteve um melhor resultado.

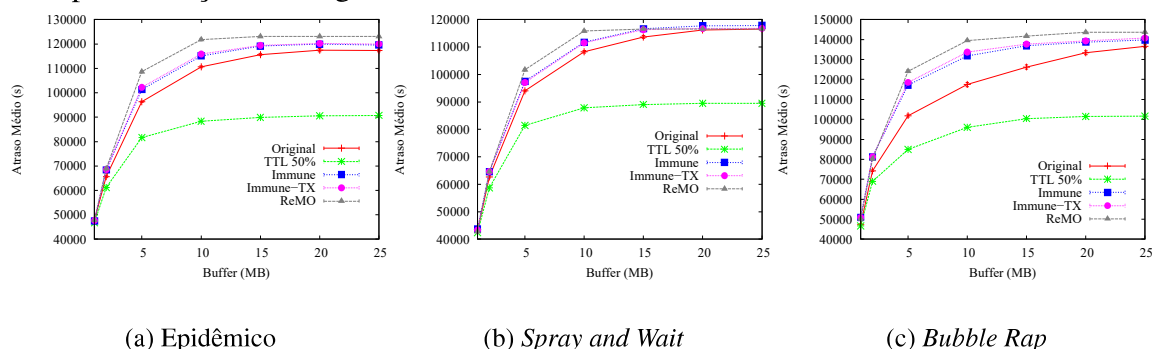


Figura 3. Atraso médio no cenário UCL1.

Na Figura 4 é analisado o atraso médio no cenário Rollernet. Na Figura 4(a) para o Epidêmico, pode-se observar que o uso do ReMO melhorou a métrica atraso médio em relação aos outros mecanismos e também ao original, neste cenário a mobilidade entre os nós é grande, aumentando os contatos e consequentemente aumentando a troca de mensagens. Como o ReMO usa os contatos para atualizar a sua lista de mensagens entregues no

destino para em seguida removê-las, por isto teve um resultado melhor que os outros mecanismos. O TTL de 50% também apresentou um resultado favorável para este cenário, perdendo somente para o ReMO. Na Figura 4(b) para o protocolo *Spray and Wait*, neste cenário com o uso do protocolo *Spray and Wait* o uso dos mecanismos não apresentaram uma melhora na métrica atraso médio, pois este protocolo não dissemina muitas mensagens na rede. Na Figura 4(c) para o *Bubble Rap*, verifica-se que neste cenário o ReMO se mantém constante para esta métrica.

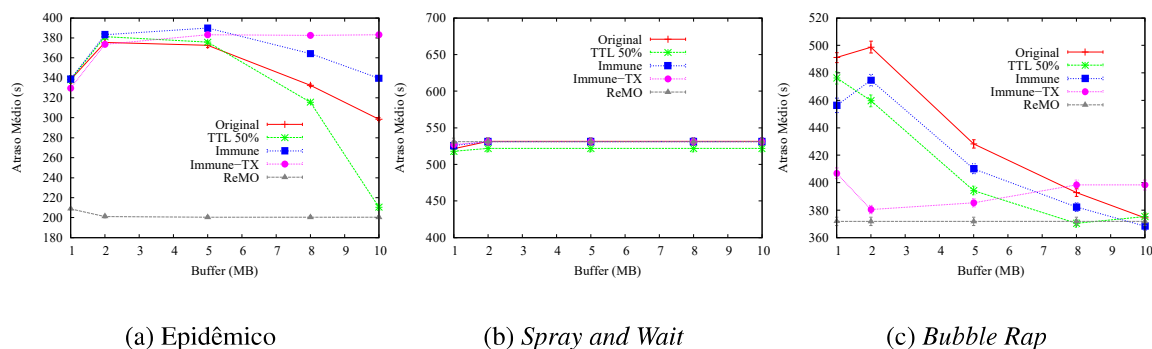


Figura 4. Atraso médio no cenário Rollernet.

#### 4.2.1. Sobrecarga de mensagens

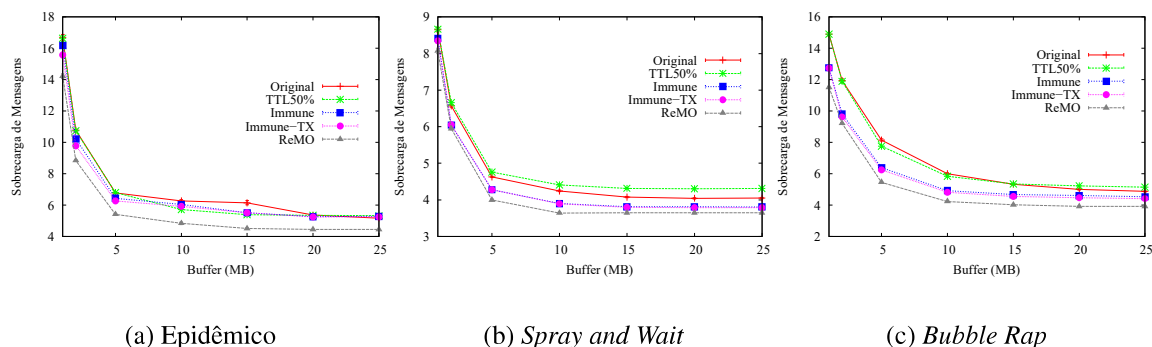


Figura 5. Sobrecarga de mensagens no cenário UCL1

Na Figura 5 é analisada a sobrecarga para o cenário UCL1. Na Figura 5(a) para o Epidêmico, verifica-se que o ReMO também apresenta uma melhora de performance para a métrica sobrecarga de mensagens. Neste caso, o protocolo de roteamento Epidêmico dissemina muitas mensagens pela rede e como o ReMO remove as mensagens obsoletas, isso reduz a quantidade de mensagens repassadas, consequentemente melhorando esta métrica. Os outros mecanismos também removem as mensagens obsoletas, mas como removem uma quantidade menor, por exemplo, o mecanismo ReMO removeu para este cenário, com *buffer* de 1MB, em média 1.648, o Immune-TX 1.454 e o Immune 1.419. Observamos que a quantidade de mensagens removidas é diretamente proporcional ao resultado obtido. Na Figura 5(b) para o *Spray and Wait*, mesmo para este protocolo que repassa um número menor de mensagens, ainda assim houve uma melhora na performance com a utilização do mecanismo ReMO e de outros mecanismos. Isto se deve ao fato da remoção das mensagens obsoletas, que evitam que mensagens já entregues no destino sejam reenviadas pela rede. Na Figura 5(c) para o *Bubble Rap*, observa-se que o mecanismo ReMO teve um resultado melhor que os outros, pois esta métrica calcula a quantidade de mensagens repassadas e divide pela quantidade de mensagens entregues.



Isto favorece o mecanismo ReMO em relação aos outros, pois como remove mais mensagens obsoletas que os outros, reduz a quantidade de mensagens repassadas na rede tendo um melhor desempenho quanto a sobrecarga.

Na Figura 6 é analisada a sobrecarga para o cenário Rollernet. Na Figura 6(a) para o Epidêmico, neste cenário temos uma grande mobilidade dos nós, isto favorece a entrega das mensagens e também a quantidade de repasses pelo uso do protocolo Epidêmico. Como os mecanismos removem mensagens já entregues no destino, possibilitando que mensagens que não são normalmente descartadas fiquem mais tempo na rede, isto faz com que estas mensagens sejam repassadas mais vezes pela rede. É possível observar que para o TTL de 50% que descarta as mensagens na metade da simulação, tem uma performance até melhor que o próprio ReMO para o *buffer* de 10MB, onde o número de mensagens repassadas é dividida pela quantidade de mensagens entregues. Na Figura 6(b) para o *Spray and Wait*, observa-se o uso dos mecanismos para este protocolo quase não sofre influência no resultado para este traço de mobilidade real. Isto se deve ao fato do protocolo *Spray and Wait* ser um protocolo replicador controlado, onde dissemina um número controlado de mensagens na rede, como este traço de mobilidade tem um grau elevado de movimentação entre os nós, fazendo com que as mensagens sejam trocadas e entregues no destino mais rapidamente. Como a sobrecarga, em sua definição, é a quantidade de mensagens repassadas dividida pelo número de mensagens entregues, por existir um controle maior na replicação das mensagens, faz com que os mecanismos tenham pouca influência na remoção, tornando-os inexpressíveis. Na Figura 6(c) para o *Bubble Rap*, observa-se que o mecanismo ReMO mantém constante a sobrecarga de mensagens em relação a todas as variações de *buffer*, isto se deve a dois fatos importantes, o primeiro é que este protocolo já remove as mensagens dos nós que não têm chance de chegar ao destino, sendo que o uso dos mecanismos diminui mais ainda o número de mensagens que são repassadas pela rede, reduzindo consideravelmente a sobrecarga de mensagens.

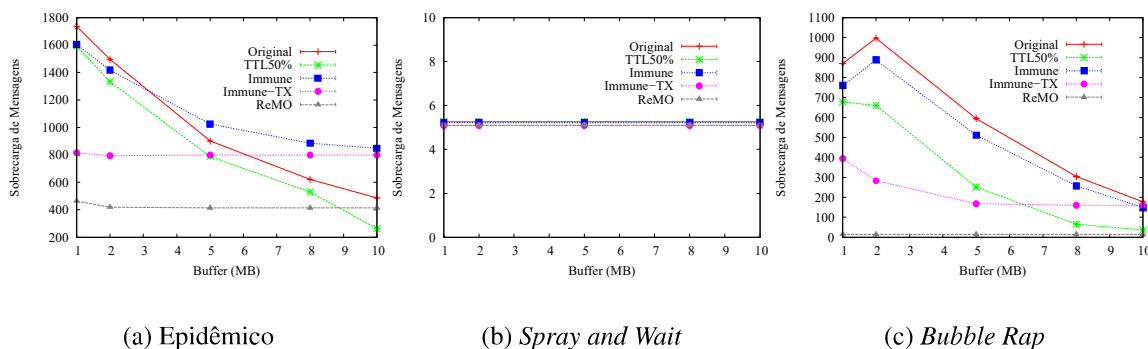


Figura 6. Sobrecarga de mensagens no cenário Rollernet.

## 5. Conclusão

Neste trabalho é apresentado uma implementação de um mecanismo de remoção de mensagens obsoletas chamado ReMO, que é semelhante ao modelo VACCINE. Além disso, foi comparado a outros dois mecanismos matemáticos citados em [Z. J. Haas 2006], *IMMUNE* e *IMMUNE-TX* juntamente com o TTL com 50% do tempo total de simulação. Todos os mecanismos implementados inclusive o ReMO, podem ser usados independente do protocolo de roteamento. É possível concluir, segundo análises realizadas através das simulações, que o mecanismo ReMO apresentou um resultado superior. Pode-se observar ainda, que mesmo usando protocolos onde o número de mensagens disseminadas é

controlado, caso do *Spray and Wait*, ou onde o protocolo utiliza o contexto social dos nós para disseminar as mensagens, caso do protocolo *BUBBLE Rap*, ele apresentou para estes dois protocolos uma melhora nas métricas avaliadas. Para o protocolo Epidêmico, que dissemina mensagens a todos os nós que tiver contato, para aumentar a chance de entrega e diminuir o atraso, também teve bons resultados com o uso do ReMO.

O uso de traços de mobilidade real tornou a simulação mais realística, pela qual se pode observar que, dependendo da mobilidade, quantidade e tempo de contato dos nós, o uso do mecanismo de remoção pode melhorar o desempenho de toda a rede.

## Referências

- A. Vahdat, D. B. (2000). Epidemic routing for partially-connected ad hoc networks. *Technical Report CS-200006*, page 18.
- Abdesslem, F. B., Henderson, T., and Parris, I. CRAWDAD trace st\_andrews/locshare/2010/uc11 (v. 2011-10-12).
- Fall, K. (2003). A delay-tolerant network architecture for challenged internets. *SIGCOMM '03*, Vol. 10(863960):27–34.
- Gomes, E. d. N., Fernandes, R. M., Campos, C. A. V., and Viana, A. C. (2012). Um mecanismo de remoção de mensagens obsoletas para as redes tolerantes a atrasos e interrupções. *WPerformance*, pages 1–14.
- Hui, P., Crowcroft, J., and Yoneki, E. (2011). Bubble rap: social-based forwarding in delay tolerant networks. *IEEE Transactions on Mobile Computing*, 10(11):1576–1589.
- Kim, M., Kotz, D., and Kim, S. (2006). Extracting a mobility model from real user traces. *INFOCOM*, pages 1–13. IEEE.
- Leguay, J. and Benbadis, F. CRAWDAD data set upmc/rollernet (v. 2009-02-02).
- Moreira, W., Mendes, P., and Sargento, S. (2012a). Assessment model for opportunistic routing. *Latin America Transactions*, 10(3):1785–1790. IEEE.
- Moreira, W., Mendes, P., and Sargento, S. (2012b). Opportunistic routing based on daily routines. *WoWMoM*, pages 1–6. IEEE.
- Rhee, I., Shin, M., Hong, S., Lee, K., Kim, S. J., and Chong, S. (2011). On the levy-walk nature of human mobility. *IEEE/ACM Transactions on Networking*, 19(3):630–643.
- S. Kaveevivitchai, H. Ochiai, H. E. (2010). Message deletion and mobility patterns for efficient message delivery in dtns. (*PERCOM Workshops*), pages 760–763.
- T. Spyropoulos, K. Psounis, C. R. (2005). Spray and wait: An efficient routing scheme for intermittently connected mobile networks. *ACM SIGCOMM (WDTN)*, pages 252–259.
- Thakur, G. S., Kumar, U., Hsu, W., and Helmy, A. (2011). Gauging human mobility characteristics and its impact on mobile routing performance. *International Journal of Sensor Networks*, 11:179–191.
- Yuen, W. H. and Schulzrinne, H. (2010). Message replication and deletion in delay tolerant networks under hop-based and time-based ttl schemes. *Citeseer*, pages 1–12.
- Z. J. Haas, T. S. (2006). A new networking model for biological applications of ad hoc sensor networks. *IEEE/ACM Transactions on Networking*, 14(1):27–40.