

Uma nova política de gerência de *buffer* para Redes Tolerantes ao Atraso e Desconexões baseada em relações sociais

Camilo Souza, Edjair Mota, Leandro Galvão, Diogo Soares, Juliano F. Naves

Resumo— Alguns tipos de Redes Tolerantes a Atraso e Desconexões precisam da ajuda dos humanos para repassar dados entre os usuários. No entanto, humanos são socialmente egoístas e tendem a querer ajudar somente usuários com quem mantém uma relação social, fato que prejudica o desempenho da rede. Diante do espaço limitado de memória secundária nos dispositivos, a gerência de *buffer* se torna fator crucial para que o desempenho da rede seja satisfatório. Este trabalho apresenta uma nova política de gerência de *buffer* para DTNs, baseada na força da relação social entre os usuários. Essa política foi avaliada em conjunto com o algoritmo de roteamento Epidêmico. Os resultados obtidos mostram que o esquema proposto aumenta a taxa de entrega além de diminuir o atraso médio e a média de saltos no cenário avaliado.

Palavras-Chave— Redes Tolerantes ao Atraso e Desconexões, relacionamento social, gerência de *buffer*.

Abstract— Some types of Delay and Disruption Tolerant Networks (DTNs) need of humans help to send data between users. However, humans are socially selfishness and prefer to help users with whom they have a social relationship, fact that affects the network performance. Given the limited memory secondary space in the devices, the buffer management becomes a crucial factor to satisfactory network performance. This paper presents a novel buffer management policy for DTNs, based on social relationship strength between users. We evaluated this policy in conjunction with the Epidemic routing algorithm. The results show that this policy increases the delivery rate besides decreasing the average delay and the average number of hops in the evaluated scenario.

Keywords— Delay and Disruption Tolerant Networks, social relationship, buffer management.

I. INTRODUÇÃO

A pilha de protocolos TCP/IP, base da arquitetura da Internet, requer premissas básicas para o seu bom funcionamento, tais como existência de uma rota fim-a-fim, baixa perda de pacotes e baixa latência. No entanto, determinados ambientes de rede não fornecem algumas dessas premissas, fato que torna a operação desses protocolos inadequada e pouco robusta em tais ambientes [3]. Exemplos de tais ambientes são as comunidades rurais, redes de sensores sem fio, redes oportunistas, entre outros. Redes Tolerantes ao Atraso e Desconexões (DTNs)

Camilo Souza, Edjair Mota, Leandro Galvão, Diogo Soares
Instituto de Computação – Universidade Federal do Amazonas (UFAM), Manaus-AM, Brasil, Emails: kmilosouza@gmail.com, edjair.galvao@icomp.ufam.edu.br, dsmoz@icomp.ufam.edu.br

Juliano Fischer Instituto de Computação/Universidade Federal Fluminense e Instituto Federal de Rondônia, Rio de Janeiro-RJ, Ji-Paraná-RO, Brasil, Email: juliano.naves@ifro.edu.br

foi o nome dado às redes de comunicação de dados que consideram estes aspectos.

Nessas redes, explora-se as eventuais conectividades criadas pela movimentação dos nós para carregar-se e transmitir-se mensagens [10], utilizando-se o mecanismo chamado de *store-carry-and-forward*. Através deste mecanismo, um nó torna-se capaz de armazenar uma mensagem no seu *buffer* de forma persistente para encaminhá-la ao destino ou a outros nós intermediários em contatos posteriores. No entanto, apesar de possibilitar a comunicação, a combinação deste mecanismo com protocolos de replicação de mensagens pode levar a muitos cenários de transbordamento do *buffer* dos nós e causar o fenômeno chamado de *buffer overflow*, o qual se refere à falta de espaço suficiente em *buffer* quando uma nova mensagem deve ser armazenada.

Em DTN, uma política de gerência de *buffer* define qual mensagem descartar em casos de *buffer overflow* [13]. Na literatura DTN existem diversas propostas de políticas para o gerenciamento do *buffer* dos nós. De acordo com [15], as políticas de gerenciamento existentes na literatura podem melhorar o desempenho de uma DTN em um determinado grau, mas todas têm suas limitações.

Neste artigo, uma nova política de gerenciamento de *buffer* para DTNs é proposta. Diferentemente de outras políticas disponíveis na literatura, esta política faz o uso de características sociais, tomando como base para o descarte de uma mensagem a força da relação social entre os nós da rede. Denominou-se esta política por *Drop Less Known* (DLK), por descartar primeiramente mensagens destinadas a usuários menos conhecidos. Utilizando um trace de mobilidade real disponível na literatura [4], uma avaliação de desempenho via simulação do algoritmo de gerência proposto foi realizada, avaliando-se métricas de entrega de mensagens. Comparada com outras políticas disponíveis na literatura, os resultados do desempenho da política proposta mostraram-se promissores.

As demais seções deste trabalho estão divididos da seguinte forma: na seção 2, são apresentados os trabalhos relacionados e na seção 3 apresenta-se a motivação e o funcionamento detalhado da política DLK. Na seção 4 são apresentados detalhes sobre os experimentos realizados e os resultados obtidos nas simulações realizadas. As conclusões e trabalhos futuros são apresentados na seção 5.

II. TRABALHOS RELACIONADOS

Na literatura sobre redes DTN, existem diversos trabalhos referentes ao gerenciamento de *buffer*. De acordo com [15] as

políticas de gerenciamento de *buffer* existentes podem ser divididas em duas categorias: políticas baseadas em informações da rede e políticas baseadas em informações locais.

As políticas baseadas em informações locais, geralmente realizam apenas uma poda simples para tomar a decisão de descarte. Exemplos de políticas pertencentes a esta classe são as seguintes: *drop random* - descarte aleatório, *drop least recently received* (LRR)- descarte da mensagem menos recentemente recebida, *drop oldest* - descarte da mensagem mais antiga no *buffer*, *drop last* - descarte da última mensagem na fila, *drop front* - descarte da primeira mensagem na fila, *drop youngest* - descarte da mensagem mais nova e *drop largest* - descarte da mensagem de maior tamanho.

Qaisar e Rashid [1] propõem uma política de gerência de *buffer* que utiliza um parâmetro chamado de *Threshold(T)* para realizar a decisão de descarte. O valor de T representa um limiar referente ao tamanho da mensagem e, em caso de *overflow*, o algoritmo primeiramente verifica se a mensagem candidata a entrar no *buffer* está dentro deste limiar. Caso não esteja, nenhuma mensagem é retirada do *buffer* e a mensagem candidata é descartada. Os resultados obtidos mostram que *T-Drop* reduz o descarte de mensagens, média de saltos e a sobrecarga na rede, enquanto aumenta a probabilidade de entrega, em comparação com a política *drop largest*.

Krifa et. al propõem em [8] uma política de gerência de *buffer* baseada na aprendizagem estatística sobre o conhecimento global da rede onde cada nó mantém uma lista de nós encontrados e o estado de cada mensagem em seu *buffer*. Entretanto o tempo de aprendizagem sobre o conhecimento global da rede pode ser elevado em DTNs, devido a volatilidade da comunicação.

Lijun TANG et. al [15] utilizaram a frequência média de encontros entre dois pares de nós para escalonar mensagens a serem enviadas e o número de réplicas de uma mensagem para tomar a decisão de descarte. Os resultados dos experimentos mostram que a política pode beneficiar a melhoria do desempenho de redes DTN em termos da taxa de entrega, atraso médio e da média de saltos.

Fischer et. al propuseram duas políticas de gerência de *buffer* em [13]. Essas políticas foram denominadas por *Less Probable Sprayed* (LPS) e *Least Recently a Forwarded* (LRF). A política LPS utiliza a quantidade de replicações de uma mensagem juntamente com a probabilidade de entrega para decidir que mensagem descartar. A política LRF, por sua vez, descarta a mensagem encaminhada a mais tempo. Os resultados obtidos por ambas as políticas LPS e LRF apresentam um aumento na taxa de entrega, além de uma menor sobrecarga da rede.

Este trabalho apresenta uma nova política de gerenciamento de *buffer* para DTNs. Diferentemente dos demais trabalhos na literatura, a política proposta utiliza como informação base para decidir que mensagem descartar, o relacionamento social entre os nós da rede.

III. GERÊNCIA DE BUFFER UTILIZANDO RELAÇÕES SOCIAIS

A consideração de características sociais forneceu um novo ângulo de visão no projeto de algoritmos para DTNs [18].

Segundo [18], grande parte dos algoritmos existentes levam em consideração somente as oportunidades de contatos dentro da rede. No entanto, existem diversas características sociais que influenciam o desempenho destes algoritmos, como por exemplo o egoísmo e as relações sociais. Recentemente, novos algoritmos de roteamento têm sido propostos, considerando algumas destas características, como em [14], [2] e [17].

A política proposta neste trabalho é baseada na ideia de que uma dessas características dos humanos influencia a escolha de que mensagem descartar em casos de *overflow*. Essa característica é o egoísmo. De acordo com [9], no mundo real a maioria das pessoas são socialmente egoístas, ou seja, estão dispostas a transmitir mensagens para pessoas com quem mantém relações sociais, mas com outras não, e tal disposição varia de acordo com a força do laço social.

Baseando-se nesses fatos, foi proposta uma política para gerência de *buffer* de DTNs baseada na força da relação social entre os usuários da rede. O principal objetivo desta política é priorizar o armazenamento de mensagens destinadas a usuários com quem se tem maior relação social e descartar mensagens destinadas a usuários menos conhecidos. Por este motivo, denominou-se essa política por *Drop Less Known* - DLK. Neste sentido, adotou-se uma nova métrica chamada de *força da relação social*, que é utilizada por DLK como base para o descarte. A seguir, maiores detalhes sobre a métrica.

A. Métrica força da relação social

O algoritmo DLK utiliza a métrica *força da relação social* para realizar o descarte de uma mensagem em caso *buffer overflow*. Essa métrica representa a força da relação social entre dois pares de nós dentro da rede. Classificou-se a relação social entre dois nós em 3 tipos: *laço forte*, *laço médio* e *laço fraco*.

Para realização de tal classificação, foram utilizadas informações retiradas do trace de mobilidade utilizado na avaliação de desempenho realizada neste trabalho, denominado *Reality* [4]. Este trace foi gerado a partir de um experimento com dispositivos *Bluetooth* realizado no *Massachusetts Institute of Technology* (MIT), utilizando um grupo de 97 usuários por aproximadamente 9 meses. Foram disponibilizadas para a comunidade científica informações de comunicação, proximidade, localização e informações de atividades destes usuários.

A principal motivação para o uso deste trace é o histórico de ligações entre os usuários que participaram dos experimentos. Esta informação é utilizada como base para classificar-se a força da relação social entre dois nós na rede. Além do histórico de ligações utiliza-se a quantidade de encontros entre os nós. Baseando-se nessas duas informações, adotou-se os seguintes critérios para classificar-se a força dos laços sociais:

- *laço forte* - é o laço entre dois nós que tem quantidade de encontros alto e que efetuaram chamadas entre si no histórico de ligações.
- *laço médio* - é o laço entre dois nós que tem quantidade de encontros alto ou médio.
- *laço fraco* - é o laço entre dois nós que não se encontram.

A seguir, maiores detalhes da política de gerência de *buffer* proposta neste trabalho.

B. Política de Gerência de *buffer Drop Less Known*

Em caso de *buffer overflow*, a política DLK inicia um procedimento de verificação para identificar que mensagem deve ser retirada do *buffer* como descrito no Algoritmo 1. No entanto, primeiramente verifica-se a força da relação social com o destino da mensagem candidata a entrar no *buffer*. Se o laço entre os dois nós for um laço fraco, DLK descarta a própria mensagem candidata. Senão, a política seleciona a mensagem presente no *buffer* destinada ao usuário menos conhecido, ou seja, aquele com quem a força do laço social é mais fraco. Se existir empate nessa escolha, a política seleciona a primeira dessas mensagens encontrada no *buffer*. Abaixo o pseudo-código da política de gerência de *buffer* DLK.

Algoritmo 1 Algoritmo Drop Less Known

```

1: procedure DLK( $m, \text{buffer}[]$ )
2:    $\text{strengthMin} \leftarrow 0$ 
3:   if  $m.\text{rel}.\text{Strength} > \text{strengthMin}$  then
4:      $\text{strengthMin} \leftarrow m.\text{rel}.\text{Strength}$ 
5:     for  $i = 1$  to  $\text{buffer}.\text{size}()$  do
6:       if  $\text{buffer}[i].\text{rel}.\text{Strength} < \text{strengthMin}$  then
7:          $\text{strengthMin} \leftarrow \text{buffer}[i].\text{rel}.$ 
8:          $\text{id} \leftarrow \text{buffer}[i].\text{id}$ 
9:       end if
10:    end for
11:  else
12:     $\text{id} \leftarrow m.\text{id}$ 
13:  end if
14: end procedure

```

A seguir, maiores detalhes sobre a avaliação de desempenho e os resultados obtidos neste trabalho.

IV. AVALIAÇÃO E RESULTADOS

Para validar a proposta de gerência de *buffer* baseada na força da relação social para DTNs, utilizou-se a técnica da simulação. Nesse trabalho implementou-se um modelo de simulação que inclui a política de gerência de *buffer* DLK e outras políticas disponíveis na literatura, permitindo assim quantificar-se o desempenho da política proposta. Esse cenário foi implementado no simulador *Opportunistic Network Environment* (ONE) [7], onde experimentos foram mensurados visando avaliar métricas de entrega de mensagens na rede.

Como citado anteriormente, utilizou-se neste trabalho um trace de mobilidade denominado *Reality* [4]. Além disso, utilizou-se como roteamento o algoritmo Epidêmico [16], o qual é baseado em replicações de mensagens, fato que aumenta a quantidade de *overflow*, beneficiando assim a avaliação da eficiência das políticas de gerenciamento do *buffer*. Demais parâmetros das simulações encontram-se na Tabela I.

Para avaliação de desempenho da política DLK, realizou-se simulações de horizonte infinito. A justificativa é muito simples: não é possível conhecer antecipadamente o tamanho da amostra que vai produzir a resposta de uma simulação com uma precisão relativa desejada. A única maneira de controlar o erro estatístico de um procedimento de estimação é analisar os

TABELA I
ALGUNS PARÂMETROS ADOTADOS NAS SIMULAÇÕES

Velocidade de Transmissão	250 kbps
Quantidade de Mensagens	1 mensagens/min
Tamanho das mensagens	500 kb

dados sequencialmente [12]. Neste trabalho, a largura do intervalo de confiança foi obtida implementando-se uma versão sequencial do método *Overlapping Batch Means* (OBM) [11], que processa um conjunto de observações da métrica de desempenho de interesse, no estado de equilíbrio dinâmico (estado estacionário). Para alcançar esse estado, implementou-se o método *Mean-Squared-Error-Reduction* (MSER-5) [6] para detectar o fim do período transiente e descartar as observações coletadas até então.

Após seguir a mesma heurística do método clássico *Nonoverlapping Batch Means*, onde é encontrado M , o tamanho ótimo de B blocos contíguos de observações cujas médias podem ser consideradas sem correlação dentro de um certo nível de significância desejado, o OBM agrupa as observações do estado estacionário em blocos de tamanho M . O bloco 1 começa na observação 1 vai até a observação M ; o segundo bloco começa na observação 2 e vai até a observação $M+1$, e assim por diante. Portanto, obtém-se $N - M + 1$ blocos. As médias desses blocos são obtidas, assim como a média global e a variância, informações necessárias para se construir um intervalo de confiança de 95%. Se o erro relativo pretendido (5%), definido como a relação entre a metade da largura do intervalo de confiança e a média global, não foi alcançado, aumenta-se o valor de M e a simulação continua coletando mais observações até obter o valor de N igual a $B.M$, quando o teste do erro relativo citado acima é feito mais uma vez. A simulação termina quando a precisão relativa desejada é alcançada.

A. Resultados

Segundo [5], o desempenho de uma DTN pode ser medido em termos da razão da taxa de entrega e do atraso médio de entrega. Dessa forma, utilizou-se três métricas na avaliação de desempenho realizada neste trabalho, a saber: taxa de entrega, atraso médio de entrega e média de saltos. A política DLK é comparada com outras três políticas de gerenciamento de *buffer* para DTNs. As políticas utilizadas nesta comparação foram *drop least recently received* (LRR), *drop Oldest* e *Evict most forwarded first* (MOFO).

1) *Taxa de Entrega*: A taxa de entrega é definida como a razão entre a quantidade de mensagens entregues ao destinatário e a quantidade de mensagens criadas.

Na Figura 1, apresenta-se o resultado da taxa de entrega no cenário *Reality*. Pode-se perceber que a política DLK obteve a melhor taxa de entrega no cenário avaliado superando as demais políticas testadas. A política LRR obteve o segundo melhor resultado, mantendo um desempenho inferior ao de DLK variando entre 8 e 10 pontos percentuais. É interessante observar que todas as políticas obtiveram um comportamento semelhante: quanto maior o tamanho do espaço em *buffer*,

maior a taxa de entrega. A justificativa para este fato é simples: quanto mais espaço, mais mensagens podem ser armazenadas e, conseqüentemente, mais mensagens poderão ser enviadas em uma oportunidade de contato, fato que aumenta a probabilidade de entrega. As políticas *drop Oldest* e *Evict most forwarded first* (MOFO) obtiveram um resultado semelhante. É interessante destacar que para tamanhos de *buffer* de 10 MB a 40 MB, a política *drop Oldest* foi superior à MOFO, mas para *buffer* superior a 40 MB, MOFO superou *drop Oldest*. No entanto em ambas as situações as diferenças variaram entre 1 e 3 pontos percentuais.

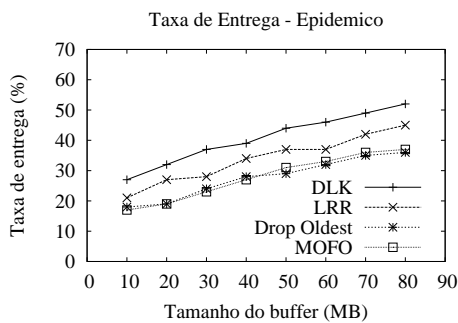


Fig. 1. Taxa de entrega utilizando roteamento Epidêmico no trace de Reality

Na Figura 2, apresenta-se a taxa de entrega de acordo com a força do laço social. Os valores apresentados neste gráfico representam a razão entre a quantidade de mensagens entregues por tipo de laço social e a quantidade de mensagens criadas por tipo de laço social. Pode-se notar no gráfico que as mensagens para usuários com quem se tem um laço mais forte são entregues em taxas bem maiores do que para os outros tipos de laços. Para tamanhos menores de *buffer* essa diferença chega a aproximadamente 45 pontos percentuais. Os dados apresentados neste gráfico denotam a vantagem de considerar-se a vontade dos usuários na gerência de *buffer* em DTNs.

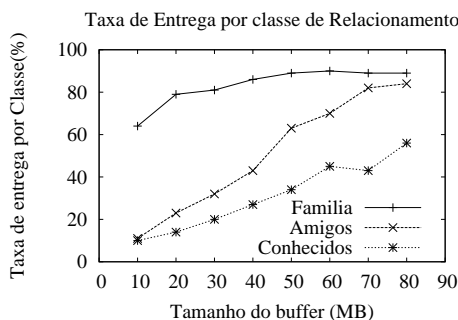


Fig. 2. Taxa de entrega de acordo com a força do laço social

A estratégia adotada pela política DLK beneficia de forma natural o aumento da taxa de entrega na rede, visto que no *buffer* sempre são mantidas as mensagens que o usuário, por algum motivo, gostaria de carregar e provavelmente entregará ao destinatário. Este fato é corroborado pelos resultados exibidos nos gráficos 1 e 2. No entanto, ao mesmo tempo que beneficia a taxa de entrega, o DLK pode aumentar o atraso

médio de entrega caso o tempo médio de encontro com os usuários que deseja-se ajudar sejam altos.

2) *Atraso médio de entrega*: O atraso médio de entrega é definido como o intervalo de tempo médio entre o envio e o recebimento das mensagens na rede. Desta forma, é importante que o atraso médio de entrega em uma DTN seja minimizado. Na Figura 3 é exibido os resultados obtidos no cenário *Reality*. A política DLK obteve o menor atraso médio para tamanhos de *buffer* a partir de 40 MB. É interessante observar que para tamanhos menores, outras políticas obtiveram os melhores resultados.

Pode-se perceber na Figura 3 um comportamento diferente do desempenho apresentado por DLK e o das demais políticas testadas, à medida que o espaço em *buffer* aumenta. No caso da política DLK, observa-se um comportamento inversamente proporcional, ou seja, ao aumentar-se o espaço em *buffer*, diminui-se o atraso médio, ocorrendo o inverso para as demais políticas.

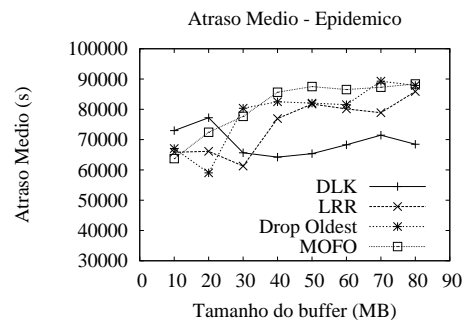


Fig. 3. Atraso Médio utilizando roteamento Epidêmico no trace de Reality

Priorizar a manutenção de mensagens no *buffer* destinadas a outros usuários com quem se tem uma relação social forte pode contribuir para a diminuição do atraso médio se o tempo médio entre encontros com estes nós for baixo. Neste sentido, o atraso médio de entrega da política DLK é fortemente influenciado por este fator, de tal forma que se o tempo médio entre encontros com os nós que deseja-se ajudar ocorrer em espaços curtos de tempo, o atraso médio de entrega será diminuído pois a mensagem será entregue. No entanto, se este tempo médio for alto, conseqüentemente o atraso médio apresentado pela política DLK também o será.

3) *Média de Saltos*: A média de saltos é definida como a quantidade média de saltos de uma mensagem até ser entregue ao destino. No cenário *Reality* avaliou-se a média de saltos para as mensagens entregues para cada uma das políticas de gerenciamento e os resultados são apresentados na Figura 4. Pode-se notar a partir dos resultados obtidos que quanto maior o espaço disponível de *buffer*, menor é a quantidade média de saltos para a maioria das políticas, entretanto, a política DLK tende a se manter estável, porém, com um leve crescimento devido a maior disponibilidade de recursos para manter as mensagens por maior tempo na rede, dando assim uma maior probabilidade de entrega para as mensagens armazenadas.

Nas demais políticas avaliadas a média de saltos possui um comportamento descendente ocasionado pela alta taxa replicação de conteúdo na rede, que com mais recursos à

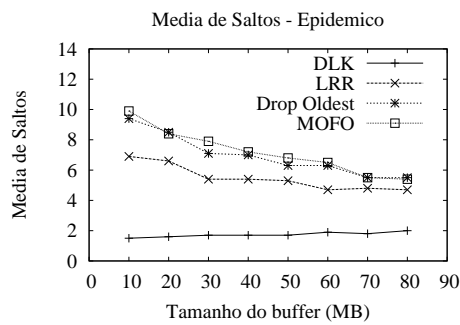


Fig. 4. Média de saltos utilizando roteamento Epidêmico no trace de *Reality*

disposição, consegue manter o conteúdo por mais tempo no *buffer*, diminuindo a quantidade média de saltos. É interessante notar que as políticas *drop Oldest* e *MOFO* obtiveram um resultado semelhante. Isso se deve as características das duas políticas, as quais tem por objetivo manter o tempo de vida da mensagem na rede afim de prover uma melhor taxa de entrega. Nota-se também que a política *LRR* tem uma média de saltos menor e com uma fração descendente, causado pela característica da política *LRR* que favorece eventualmente contatos mais frequentes entre pares de nós.

Aliado às replicações de mensagens feitas pelo algoritmo Epidêmico, a política *DLK* realiza uma poda no *buffer*, dando prioridade a mensagens que o usuário gostaria de carregar e que tem assim uma probabilidade de entrega maior. Neste sentido, a estratégia da política *DLK* diminui a quantidade média de saltos necessárias para uma mensagem chegar ao destino.

V. CONCLUSÕES

Neste trabalho, apresentou-se uma política de gerência de *buffer* para redes DTN baseada em relações sociais. Partindo-se da suposição de que no mundo real os usuários são socialmente egoístas e desejam ajudar usuários de acordo com a força da relação social entre eles, o objetivo principal da política, é indicar, no caso de *overflow*, mensagens destinadas a outros usuários com quem se tem uma relação social mais fraca.

A política proposta obteve uma maior taxa de entrega, alcançando até 15% mais mensagens entregues que a política de segundo melhor desempenho, com um menor atraso médio de entrega e com uma menor quantidade de saltos.

Pode-se destacar duas contribuições principais com a criação da política *DLK*: primeiramente, atender ao perfil egoísta dos usuários reais, fato que representa uma escolha mais correta por respeitar a vontade do usuário sobre que mensagem descartar em casos de *overflow*. Segundo, a política *DLK* utiliza como base para o descarte de uma mensagem características sociais, fator este que tem sido bastante discutido com relação ao roteamento de mensagens, mas, com relação a gerência de *buffer* para DTNs ainda não havia sido abordado. Com este trabalho pretende-se chamar a atenção da comunidade científica para discutir-se o uso de características sociais também na gerência do *buffer* para DTNs.

Em trabalhos futuros, pretende-se investigar-se o hibridismo entre métodos existentes na literatura e o método *DLK*, investigar-se a utilização de outras características sociais como as classes sociais e o altruísmo no gerenciamento de *buffer*, investigar-se as influências dos variados cenários de redes DTN no desempenho de *DLK* e dos principais algoritmos para gerência de *buffer* existentes, além de buscar-se alternativas para otimizar o método *DLK*.

REFERÊNCIAS

- [1] Qaisar Ayub and Sulma Rashid. T-drop: An optimal buffer management policy to improve qos in dtn routing protocols. *Journal of Computing*, 2(10):46–50, 2010.
- [2] Po-An Chen and David Kempe. Altruism, selfishness, and spite in traffic routing. In *Proceedings of the 9th ACM conference on Electronic commerce*, pages 140–149. ACM, 2008.
- [3] Carina T de Oliveira, Marcelo DD Moreira, Marcelo G Rubinstein, Luís Henrique MK Costa, and Otto Carlos MB Duarte. Redes tolerantes a atrasos e desconexões. *SBRC Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, 2007.
- [4] Nathan Eagle and Alex (Sandy) Pentland. CRAWDAD data set mit/reality (v. 2005-07-01). Downloaded from <http://crawdad.cs.dartmouth.edu/mit/reality>, July 2005.
- [5] G Fathima and RSD Wahidabanu. Buffer management for preferential delivery in opportunistic delay tolerant networks. *International Journal of Wireless & Mobile Networks (IJWMN)*, 3(5):15–28, 2008.
- [6] William W Franklin and KP White. Stationarity tests and mser-5: exploring the intuition behind mean-squared-error-reduction in detecting and correcting initialization bias. In *Simulation Conference, 2008. WSC 2008. Winter*, pages 541–546. IEEE, 2008.
- [7] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.
- [8] Amir Krifa, C Baraka, and Thrasyvoulos Spyropoulos. Optimal buffer management policies for delay tolerant networks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on*, pages 260–268. IEEE, 2008.
- [9] Qinghua Li, Sencun Zhu, and Guohong Cao. Routing in socially selfish delay tolerant networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.
- [10] Yun Li, Ling Zhao, Zhanjun Liu, and Qilie Liu. N-drop: congestion control strategy under epidemic routing in dtn. In *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pages 457–460. ACM, 2009.
- [11] Marc S Meketon and Bruce Schmeiser. Overlapping batch means: Something for nothing? In *Proceedings of the 16th conference on Winter simulation*, pages 226–230. IEEE Press, 1984.
- [12] Edjair de Souza Mota. *Performance of Sequential Batching-based Methods of Output Data Analysis in Distributed Steady-state Stochastic Simulation*. PhD thesis, Universitätsbibliothek, 2002.
- [13] Igor M. Moraes Celio Vinicius N. de Albuquerque Naves, Juliano F. Lps e Irf. Políticas de gerenciamento de buffer eficientes para redes tolerantes a atrasos e desconexões. *Simpósio Brasileiro de Redes de Computadores (SBRC'12)*, pages 293–305, 2012.
- [14] Samir Okasha. Altruism, group selection and correlated interaction. *The British journal for the philosophy of science*, 56(4):703–725, 2005.
- [15] Lijun TANG, Yi CHAI, Yun LI, and Binbin WENG. Buffer management policies in opportunistic networks. *Journal of Computational Information Systems*, 8(12):5149–5159, 2012.
- [16] Amin Vahdat, David Becker, et al. Epidemic routing for partially connected ad hoc networks. Technical report, Technical Report CS-200006, Duke University, 2000.
- [17] Kuang Xu, Pan Hui, Victor O. K.Li, Jon Crowcroft, Vito Latora, and Pietro Lio. Impact of altruism on opportunistic communications. In *Proceedings of the first international conference on Ubiquitous and future networks, ICUFN'09*, pages 153–158, Piscataway, NJ, USA, 2009.
- [18] Ying Zhu, Bin Xu, Xinghua Shi, and Yu Wang. A survey of social-based routing in delay tolerant networks: positive and negative social effects. 2012.