

## Resenha do artigo “DTN-Based Data Aggregation for Timely Information Collection in Disaster Areas”

30 de maio

Professor: Alfredo Goldman

Aluno: Rafael Santos Coelho

### Resumo

Neste trabalho, fazemos uma resenha do artigo intitulado *DTN-Based Data Aggregation for Timely Information Collection in Disaster Areas*, escrito por Jovilyn Fajardo, Keiichi Yasumoto, Naoki Shibata, Weihua Sun e Minoru Ito e publicado, em 2012, nos anais da oitava edição da *IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*.

## 1 Introdução

Desastres de largas proporções, previsíveis ou não, além de ocasionar elevado número de óbitos e, por vezes, estragos ambientais preocupantes, compartilham, em geral, uma característica bem distinta: o enorme dano causado em infraestruturas de telecomunicação. Falhas parciais ou totais de tais infraestruturas implicam não somente em prejuízo financeiro, mas, pior do que isso, também impossibilitam a coleta eficiente e periódica de informações precisas acerca das áreas afetadas, o que é de vital relevância jornalística, humanitária e estratégica. Segundo os registros do *Emergency Events Database*<sup>1</sup>, entre os anos de 2002 a 2013, cerca de 4500 desastres foram catalogados. Para exemplificar, apenas nos três últimos anos, o mundo assistiu atônito às notícias do terremoto no Haiti, das enchentes no Paquistão e da tsunami que atingiu a costa leste do Japão.

Relatórios de avarias [Townsend & Moss, 2005] provocadas por grandes desastres apontam normalmente duas principais causas para colapsos de infraestruturas de telecomunicação: a destruição física de componentes da rede e o congestionamento na rede. Para lidar com essa problemática, é de suma importância o projeto de algum tipo de rede *ad-hoc* móvel ou *mobile ad-hoc network* (MANET) capaz de prover acesso a informações sobre as áreas atingidas. É sabido, porém, que o fluxo de informação (e, portanto, a confiabilidade) em uma MANET pode ser comprometido por eventuais

---

<sup>1</sup><http://www.emdat.obe>

períodos de desconectividade entre os nós, sobretudo no contexto de áreas atingidas por desastres. Dessa forma, em redes nas quais o roteamento fim-a-fim não é garantido, a arquitetura de redes tolerantes a falhas ou *delay-tolerant networks* (DTN) se mostra promissora.

DTNs são redes projetadas para operar de maneira efetiva em ambientes extremos, conhecidos pela presença de interferência e latência, usualmente medida em minutos, horas ou até dias. Uma DTN interconectando *smartphones* de pessoas já presentes nas áreas afetadas vem a ser uma solução interessante, dada sua imensa popularização e os notáveis avanços no que diz respeito ao poder de processamento, armazenamento e comunicação. Entretanto, cabe salientar que mesmo essa idéia possui suas limitações, pois sempre foi e continua sendo consensual a noção de que a capacidade limitada de armazenamento e o consumo de energia são pontos francos de telefones celulares. Dessa forma, no cenário de uma área atingida por um desastre, faz-se necessário contemplar alguma estratégia que procure reduzir não só a latência com que as mensagens são entregues, bem como a redundância dos dados.

Uma frente de solução para esse impasse se dá via o emprego de algum método de agregação de dados. Em linhas gerais, agregação de dados em MANETs consiste em agrupar inteligentemente pacotes em rota vindos de diferentes fontes e destinados a um mesmo nó. Com isso, o ganho a longo prazo inclui o desoneração da rede e, por conseguinte, seu melhor funcionamento.

Neste texto, apresentamos uma resenha do artigo *DTN-Based Data Aggregation for Timely Information Collection in Disaster Areas* [Fajardo et al., 2012]. Os autores introduzem e avaliam experimentalmente um algoritmo de agregação de dados em DTNs com o intuito de viabilizar a coleta de informações sobre uma área de interesse ou *area of interest* (AOI) localizada nas redondezas de alguma zona de desastre.

O restante deste artigo está organizado como descrito a seguir. Na Seção 2, discutiremos brevemente sobre alguns trabalhos relacionados. Na Seção 3, definimos formalmente o problema da agregação de dados. Apresentamos em alto nível, na Seção 4, o algoritmo proposto. Na Seção 5, comentamos sobre os resultados experimentais alcançados. Por fim, na Seção 6, fazemos nossas considerações acerca do trabalho em questão.

## 2 Revisão bibliográfica

Estudos recentes que abordam o uso de DTNs na coleta e transmissão de informações sobre áreas de desastres se dividem essencialmente em três vertentes, comentadas de maneira sucinta na sequência.

A primeira vertente reúne trabalhos que se baseiam na pré-implantação de dispositivos de transporte de mensagens. Nessa linha, destacamos o trabalho [Zhou et al., 2010], no qual os autores utilizam as chamadas redes de pombos (*pidgeon networks*), DTNs que exploram veículos aéreos rádio-controlados, designados de pombos, para monitoramento e coleta de informações sobre áreas de desastres.

Os pombos partem de uma única estação, visitam uma área de interesse, colhem informações captadas por sensores instalados e retornam à estação. A rota descrita por um pombo é calculada com o objetivo de minimizar a latência das mensagens geradas. Devido ao número reduzido de pombos, é instaurada alguma política de escalonamento para melhor coordená-los e, assim, reforçar o uso eficiente dos mesmos. O artigo discutido compara empiricamente duas políticas de escalonamento: sob demanda (*on demand*) e baseado em espera (*waiting-based*).

A segunda vertente se fundamenta em trabalhos que propõem abordagens parcialmente centralizadas a fim de evitar congestionamento e determinar a topologia da rede. Os artigos que selecionamos desse grupo foram [George et al., 2010; Luqman et al., 2011]. Dado o amplo escopo do artigo [George et al., 2010], restringimo-nos a comentar que, em tal trabalho, os autores propõem o que cunharam com o nome de *Distressnet*, a saber uma complexa arquitetura de rede *ad-hoc* sem fio extensível e escalável, com sensoriamento colaborativo distribuído, roteamento via protocolos *topology-aware* oportunistas e tolerantes a atrasos e tratamento de congestão por comunicação multi-canais.

Já em [Luqman et al., 2011], os autores discutem um arcabouço com mecanismos QoS (*quality of service*) que incorporam informações de contexto usando monitoramento pró-ativo de *buffers*, controle centralizado e *hop-by-hop* de congestão e ajuste automático de taxa de transmissão baseado no *feedback* da rede.

A terceira e última vertente abarca trabalhos que introduzem aplicativos de *smart-phones* para gerência de serviços de prestação de socorro a zonas de desastre. Nessa frente, citamos o artigo [Fajardo & Oppus, 2010], no qual os autores apresentam o *MyDisasterDroid*, um aplicativo para a plataforma *Android* da *Google* projetado para auxiliar equipes de ajuda humanitária. O programa combina algoritmos genéticos que buscam encontrar trajetórias curtas e seguras para o resgate de sobreviventes e fornecimento de provisões.

### 3 O problema da agregação de dados

Considere o cenário de um desastre nas primeiras horas da fase de recuperação. É fato que, durante esse período, informações acerca das necessidades da área afetada

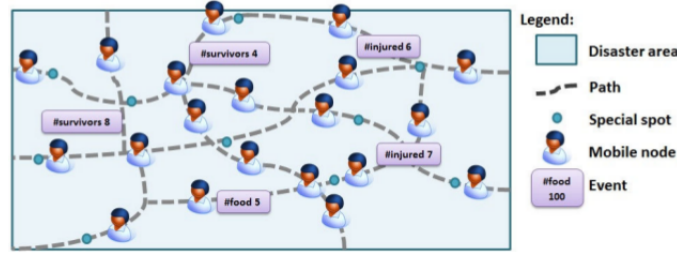


Figura 1: Exemplo de uma área de desastre e unidades móveis reportando vários eventos dentro da AOI. (Fonte: [Fajardo et al., 2012].)

precisam estar atualizadas e à disposição. Conforme discutido até aqui, sabemos que isso nem sempre ocorre por conta da ruptura das infraestruturas locais de telecomunicação.

Antes de proceder com a formalização do problema da agregação de dados, precisamos enumerar as hipóteses assumidas no trabalho em questão e introduzir alguma notação matemática. Sejam  $A_d$  e  $AOI$ , onde  $AOI \subseteq A_d$ , a área de desastre e a área de interesse, respectivamente. Chamamos de *unidade móvel* qualquer agente comunicável e presente na  $AOI$ . Fora da  $AOI$ , definimos uma unidade móvel especial  $s$ , chamada de *sorvedouro*, que gera uma requisição de coleta de informações sobre  $AOI$  no instante de tempo  $t_0$  e com uma *granularidade de agregação*  $\Phi$ , termo a ser explanado mais detalhadamente no parágrafo adiante.

No perímetro da  $AOI$ , pode haver uma ou mais unidades móveis. Cada unidade móvel reporta para o sorvedouro, por meio de uma mensagem, algum *evento* relevante (por exemplo, a descoberta de sobreviventes ou de alimento) que ocorreu na  $AOI$ . Toda *mensagem*  $m$  tem uma área circular de cobertura  $m.a$  e um raio  $r_{msg}$ . O parâmetro  $\Phi$ , mencionado anteriormente, age como um filtro. Quanto menor seu valor, mais “refinadas” precisam ser as mensagens colhidas. Em outras palavras, a granularidade de agregação é um instrumento de controle do nível de detalhes requerido acerca das informações obtidas da  $AOI$ . Além disso, dizemos que  $\Phi$  possui uma área circular  $\Phi.a$  de raio  $\Phi.r$ .

Dizemos que uma mensagem é *atômica* se ela ainda não foi agregada a nenhuma outra mensagem. Enunciamos agora a primeira restrição do problema de agregação de dados, a chamada *restrição de agregação de mensagens*. Duas mensagens  $m_1$  e  $m_2$  são agregadas se, e somente se, (i) as áreas cobertas por  $m_1$  e  $m_2$  se interceptam, (ii) a união das áreas cobertas por  $m_1$  e  $m_2$  está inteiramente contida no círculo induzido por  $\Phi$  e, por fim, (iii) a distância entre as unidades móveis que geraram  $m_1$  e  $m_2$  ultrapassa  $r_{msg}$ . A Figura 2 ilustra a situação descrita até aqui. Nela, vemos a granularidade de

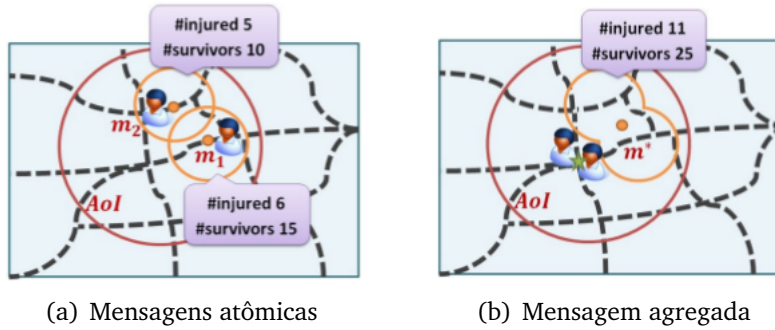


Figura 2: Exemplo de agregação de mensagens. (Fonte: [Fajardo et al., 2012].)

agregação (circunferência vermelha) e duas unidades móveis que geram mensagens atômicas (circunferências laranjas) que são, posteriormente, agregadas.

Fazemos a suposição de que cada unidade móvel desempenha um de três papéis específicos: *pessoa comum*, *socorrista* ou *oficial*. Pessoas comuns são unidades móveis que agem segundo a tendência de se afastarem de pontos de eventos, isto é, localidades dentro da AOI onde há algum relato de evento. Socorristas se comportam contrariamente às pessoas comuns e oficiais manifestam um comportamento híbrido entre pessoas comuns e socorristas.

Cada unidade móvel pode ter contato direto ou indireto com outra unidade móvel. O contato indireto entre unidades móveis é definido como o fecho transitivo do contato direto. O contato direto, por sua vez, está permitido apenas se as unidades móveis estiverem a uma distância não maior do que  $r_{\text{con}}$  uma da outra. Mais ainda, uma unidade móvel pode executar quatro atividades, a saber *criar*, *enviar*, *receber* e *agregar* mensagens. Convencionamos que toda unidade móvel tem uma posição  $pos(t)$  no instante de tempo  $t$  (possivelmente determinada por *GPS*) e que a comunicação entre duas unidades móveis  $s$  e  $s'$  se dá apenas dentro de um intervalo de tempo fixo  $cd(s, s', t)$  a partir do instante  $t$ . Em suma, a chamada *restrição de comunicação* imposta para unidades móveis permite que duas unidades móveis se comuniquem diretamente a partir de um instante  $t$  se, e somente se, (i) estiverem numa distância máxima  $r_{\text{con}}$  uma da outra e (ii) dentro do intervalo de tempo  $cd(s, s', t)$ .

Durante a execução do algoritmo, supõe-se que o tempo está discretizado em janelas de comprimento  $P$ . Cada janela se divide em dois intervalos, a saber o intervalo ativo, com duração  $zP$  e o de sono, com duração  $(1 - z)P$ , onde  $0 < z < 1$  é um parâmetro fixo previamente. Durante o intervalo ativo, a unidade móvel tenta sondar sua vizinhança a procura de outras unidades móveis. Caso nenhum vizinho tenha sido encontrado, a unidade móvel hiberna durante o intervalo de sono a fim

#### DESCOBERTA DA VIZINHANÇA

```
1   $s.V \leftarrow \emptyset \triangleright$  conjunto de vizinhos de  $s$ 
2  while  $s$  estiver em intervalo ativo
3      do  $s$  envia broadcast
4          if  $s$  receber resposta de outra unidade móvel  $p$ 
5              then  $s.V \leftarrow s.V \cup \{p\}$ 
6               $s$  e  $p$  sincronizam seus relógios
```

Figura 3: Pseudo-código do algoritmo DESCOBERTA DA VIZINHANÇA.

de poupar energia. Por fim, assumimos que cada unidade móvel possui um relógio preciso o suficiente que a permite se manter sincronizada com seus vizinhos.

Segue, então, a definição do problema da agregação de dados: minimizar o tempo para que o sorvedouro receba a mensagem agregada de resposta cobrindo toda a AOI e satisfazendo as restrições de agregação e de comunicação.

## 4 O algoritmo

Nesta seção, descrevemos resumidamente o algoritmo proposto pelos autores do artigo sob análise. A execução de tal algoritmo é disparada apenas nas unidades móveis localizadas dentro da AOI e iniciada assim que a unidade móvel recebe a requisição do sorvedouro. O algoritmo consiste em um grande laço que é executado enquanto a unidade móvel hospedeira não receber um pacote de confirmação (ACK) do sorvedouro ou não tiver ocorrido um estouro de tempo (*timeout*). O algoritmo possui quatro fases, executadas precisamente nessa ordem: *descoberta da vizinhança*, *criação da mensagem*, *troca de mensagens* e *agregação*. Transcrevemos nas Figuras 3, 4, 5 e 6 os pseudo-códigos das quatro fases do algoritmo.

Convém salientar que a agregação das mensagens feita pelo algoritmo AGREGAÇÃO usa uma estrutura de dados especial conhecida como *filtro de Bloom*. Projetados por Burton Howard Bloom em 1970, filtros de Bloom são uma estrutura de dados probabilística normalmente utilizada para implementar conjuntos e que suporta dois tipos de operações, a saber inserção e teste de pertinência. Em baixo nível, um filtro Bloom consiste de um arranjo  $X$  de  $m$  bits e  $k$  funções *hash* diferentes, cada uma das quais capaz de mapear um elemento para qualquer posição de  $X$  com probabilidade uniforme. Para inserir um elemento  $i$  no conjunto, obtemos as  $k$  chaves *hash* de  $i$  e ligamos os

#### CRIAÇÃO DAMENSAGEM

```
1  ▷ estamos supondo que  $s.V \neq \emptyset$ 
2  while  $s$  estiver em intervalo de sono
3      then  $s$  cria mensagem  $m$  ▷ se  $s$  possui algum evento para reportar
4       $s$  agrega  $m$  e  $M_s$  ▷  $M_s$  é o conjunto de mensagens criadas por  $s$ 
```

Figura 4: Pseudo-código do algoritmo CRIAÇÃO DAMENSAGEM.

#### TROCA DEMENSAGENS

```
1  ▷ estamos supondo que  $s.V \neq \emptyset$ 
2   $s$  calcula  $s.ert$  ▷  $s.ert$  é o tempo esperado para  $s$  alcançar o sorvedouro
3  Seja  $p$  a unidade móvel de menor  $ert$  em  $s.V$ 
4  if  $s.ert < p.ert$ 
5      then  $s$  retém  $M_s$  e recebe  $M_p$  de  $p$ 
6      else  $s$  envia  $M_s$  para  $p$ 
```

Figura 5: Pseudo-código do algoritmo TROCA DEMENSAGENS. (O tempo esperado ou *ert* (*expected reaching time*) para uma unidade móvel  $s$  alcançar o sorvedouro é calculado a partir da posição do sorvedouro, de  $s$  e da velocidade de  $s$ .)

#### AGREGAÇÃO

```
1  ▷ estamos supondo que  $s$  recebeu de uma unidade móvel  $p$  a mensagem  $M_p$ 
2  for all  $m_s \in M_s$  and  $m_p \in M_p$ 
3      do if  $(m_s.a \cap m_p.a \neq \emptyset)$  and  $\text{RAIO}(m_s.a \cup m_p.a) \leq \Phi.r$ 
4          then if  $m_s$  e  $m_p$  são atômicas
5              then if  $\text{dist}(p, s) > r_{\text{con}}$ 
6                  then agregar  $m_s$  e  $m_p$ 
7          else agregar  $m_s$  e  $m_p$ 
```

Figura 6: Pseudo-código do algoritmo AGREGAÇÃO.

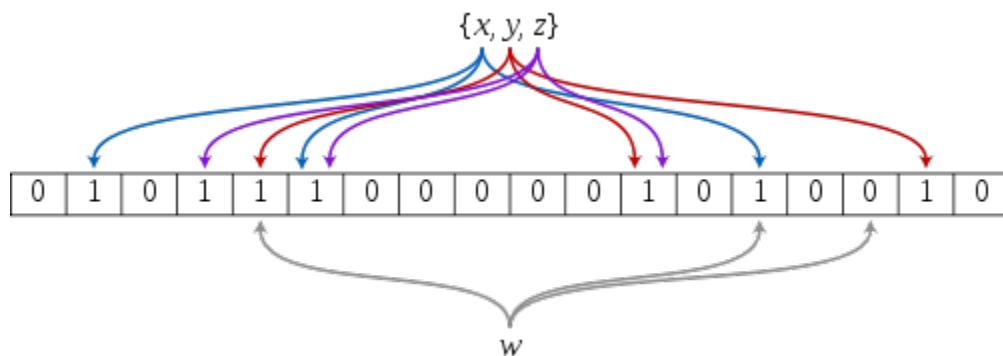


Figura 7: Filtro de Bloom no momento da inserção dos elementos  $x, y$  e  $z$  e do teste de pertinência de um quarto elemento  $w$ . (Fonte: Wikipedia.)

bits de  $X$  nas respectivas posições marcadas pelas chaves calculadas. Para testar a pertinência de um dado elemento  $i$  ao conjunto, obtemos as  $k$  chaves *hash* de  $i$  e verificamos o valor dos *bits* de  $X$  nas posições designadas pelas chaves calculadas. Se, e somente se qualquer uma dessas posições tiver um *bit* desligado, o elemento definitivamente não pertence ao conjunto. Sendo assim, é possível notar que filtros Bloom estão sujeitos a falsos positivos, isto é, um filtro de Bloom pode retornar erroneamente que um determinado elemento pertence ao conjunto. Para mitigar a incidência de falsos positivos, é preciso escolher bem o valor de  $m$  e as  $k$  funções *hash*. Na Figura 7, vemos uma ilustração de um filtro de Bloom no momento da inserção de três elementos  $x, y$  e  $z$  e do teste de pertinência de um quarto elemento  $w$ .

## 5 Avaliação experimental

O algoritmo projetado no artigo-base foi testado pelos autores por meio de uma simulação. O local escolhido para emular a área de desastre foi um lote da Takayama Sciente Town, uma parte do campus do Instituto Nara de Ciência e Tecnologia, no Japão. A área tem um perímetro quadrangular e área de 25 quilômetros quadrados. Na Figura 8, podemos ver um mapa das estradas da área descrita. Na simulação, unidades móveis foram posicionadas aleatoriamente dentro da área de desastre. Elas se locomovem com velocidade constante aleatória pelas estradas de um ponto a outro tomando rotas ditadas pelo algoritmo de Dijkstra. AAOI foi fixada com um raio de 500 metros e a granularidade de agregação  $\Phi$  com um raio de 50 metros. Foram feitas ao todo 10 simulações e calculadas médias aritméticas de algumas métricas de interesse mostradas a seguir.



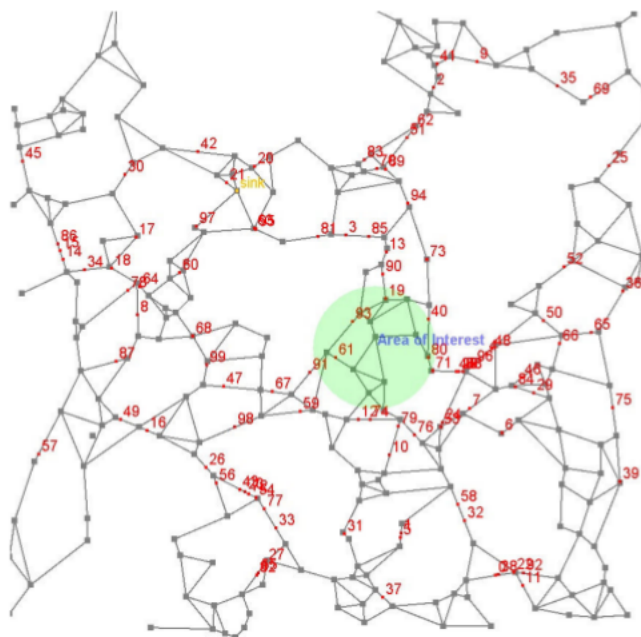


Figura 8: Ambiente de simulação. (Fonte: [Fajardo et al., 2012].)

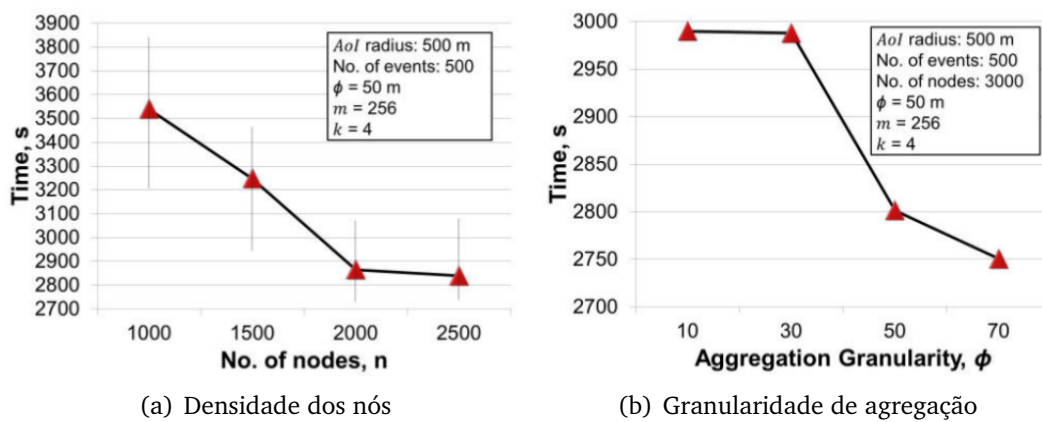


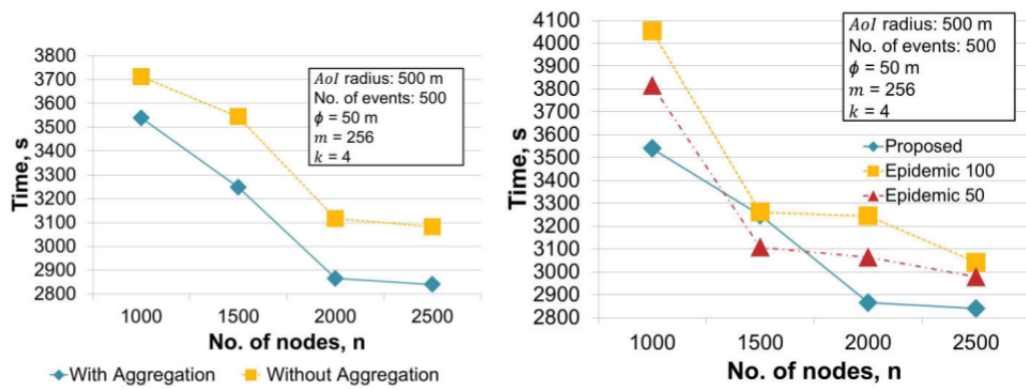
Figura 9: (Fonte: [Fajardo et al., 2012].)

Na Figura 9, mostramos gráficos com resultados de dois experimentos. No gráfico da esquerda, vemos o impacto do número de nós presentes na AOI na latência da mensagem de resposta que o sorvedouro recebe. O gráfico deixa claro que quando maior o número de unidades móveis, mais tende a cair a latência. Esse fenômeno não é surpreendente, uma vez que, com um número elevado de nós na AOI, há mais oportunidades de contato entre eles e, portanto, mais troca de mensagens, o que implica, em última análise, na cobertura rápida da AOI. Ainda na Figura 9, no gráfico da direita, vemos o impacto que o valor da granularidade de agregação tem na eficiência do algoritmo. Para valores cada vez maiores de  $\Phi$ , a curva indica que o eficiente leva menos tempo para emitir a mensagem de resposta para o sorvedouro. Tal constatação, novamente, não é demasiado impressionante, pois, como já discutido, valores elevados do parâmetro  $\Phi$  implicam em uma menor exigência de detalhes sobre os eventos ocorridos na AOI e, por consequência, em mais facilidade de cobertura da área.

Na Figura 10, vemos outros gráficos com resultados de mais dois experimentos. No gráfico da esquerda, vemos curvas que expressam o impacto da agregação de dados na eficiência do algoritmo. A curva azul é referente à versão do algoritmo com agregação de dados. A curva amarela é referente à versão do algoritmo sem agregação de dados. É notável que a curva azul domina a curva amarela. Uma explicação para esse fato se traduz na observação de que, com agregação, a tensão na rede, isto é, a carga de trabalho desempenhado por cada unidade móvel, é obviamente reduzida, visto que cada nó gasta menos tempo transmitindo dados e, dessa maneira, possui mais tempo a sua disposição para varrer a AOI. Ainda na Figura 10, vemos no gráfico da direita um experimento feito com o objetivo de testar a estratégia de roteamento do algoritmo proposto. Os autores compararam o algoritmo com duas versões do protocolo de roteamento epidêmico, a saber roteamento epidêmico com 50% de chance de repasse de mensagem e roteamento epidêmico com 100% de chance.

## 6 Considerações finais

Neste texto, fizemos uma resenha crítica do artigo *DTN-Based Data Aggregation for Timely Information Collection in Disaster Areas* [Fajardo et al., 2012]. O trabalho é bastante interessante e ambicioso. Os autores tentaram atacar um problema extremamente complexo e relevante. Julgamos que a modelagem feita por eles foi satisfatória, mas o artigo, textualmente falando, carece de um pouco de formalismo matemático. Além disso, fica a pergunta de por que os autores não testaram seu algoritmo contra outros protocolos de roteamento. Como perspectivas de trabalhos futuros, vale registrar que os autores pretendem fazer testes reais (não-simulados) com o algoritmo,



(a) Comparação entre agregação e não-agregação

(b) Comparação com roteamento epidêmico

Figura 10: (Fonte: [Fajardo et al., 2012].)

investigar possibilidades de melhorias na sua implementação e derivar algum limitante teórico para sua complexidade assintótica.

## 7 Referências

1. S. George, W. Zhou, H. Chenji, M. Won, Y. O. Lee, A. Pazarloglou, R. Stoleru, e P. Barooah, “Distressnet: a wireless ad hoc and sensor network architecture for situation management in disaster response,” *Communications Magazine*, IEEE, vol. 48, n. 3, pp. 128–136, 2010
2. F. Luqman, F. T. Sun, H. T. Cheng, S. Buthpitiya, e M. Griss, “Prioritizing data in emergency response based on context, message content and role,” in *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, ser. ACWR ’11. New York, NY, USA: ACM, 2011, pp. 63–69
3. A. M. Townsend e M. L. Moss, “Telecommunications Infrastructure in Disasters: Preparing Cities for Crisis Communications”, The Center for Catastrophe Preparedness and Response New York University, Tech. Rep., 2005
4. J. T. B. Fajardo, K. Yasumoto, N. Shibata, W. Sun; M. Ito, M., “DTN-based data aggregation for timely information collection in disaster areas,” *2012 IEEE 8th International Conference Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 333–340, 2012
5. S. M. George, W. Zhou, H. Chenji, W. Myounggyu, L. Y. Oh, A. Pazarloglou, R. Stoleru, P. Barooah, “DistressNet: a wireless ad hoc and sensor network architecture for situation management in disaster response,” *Communications Magazine*, IEEE , vol. 48, n. 3, pp. 128–136, 2010
6. T. B. J. Fajardo, C. M. Oppus, “A mobile disaster management system using the android technology”, *WTOC 9*, pp. 343–353, 2010