

Disseminação Bio-Inspirada de Eventos em Redes Dinâmicas e Descentralizadas

Adam S. Banzi, Aurora T. R. Pozo, Elias P. Duarte Jr.

Departamento de Informática – Universidade Federal do Paraná (UFPR)
Caixa Postal 19018, Curitiba PR CEP 81531-980 Brasil

{adam,aurora,elias}@inf.ufpr.br

Resumo. *Este trabalho apresenta uma estratégia baseada em inteligência coletiva para a disseminação de eventos em redes dinâmicas e descentralizadas. Um evento corresponde à mudança do estado de um nodo ou enlace da rede. Através de formigas, que são abstrações para agentes móveis, as informações são disseminadas na rede. Um nodo que detecta o evento na sua vizinhança dispara a disseminação, que é controlada por feromônios. O nível de feromônios é utilizado para controlar o tamanho da população de formigas e determinar as rotas que percorrem. Um estudo empírico foi realizado, comparando a estratégia proposta com a disseminação por inundação e gossip. Resultados permitem observar que o algoritmo apresenta uma solução de compromisso entre o tempo necessário para a disseminação e a sobrecarga em termos do número de mensagens utilizadas.*

Abstract. *This work presents a strategy based on swarm intelligence for spreading events in dynamic and decentralized networks. An event is defined as a state transition of a node or link. By means of ants, which correspond to mobile agents, information is spread throughout the network. A node which detects an event in its neighborhood starts disseminating the new information. Pheromones are used to both control the ant population and help define the paths that the agents take. An empirical study was performed, in which the proposed strategy was compared with flooding and gossip algorithms. Results obtained show that the proposed strategy presents a good trade-off between the time required to disseminate information and the overhead in terms of the number of messages.*

1. Introdução

A disseminação de eventos é um dos desafios das redes dinâmicas, tais como redes móveis *ad hoc* e *peer-to-peer*. Em modelos tradicionais de rede, com pouca modificação da composição, o controle e administração dos recursos são, comparativamente, simplificados [Dong et al. 2006]. Por outro lado, nas redes dinâmicas a própria frequência com que nodos entram e saem é alta, impedindo que as decisões de disseminação sejam tomadas de forma antecipada a partir de um conhecimento prévio da rede [Ahmed and Far 2007]. Outra característica relevante é a descentralização, ou seja, a ausência de um ponto único que possa ser responsabilizado pelo controle. Devido ao alto dinamismo, a manutenção do funcionamento da rede deve ser realizada conjuntamente pelos nodos ativos.

Para auxiliar na tarefa de gerenciamento de recursos envolvidos, inclusive na disseminação de informações, é importante que os integrantes da rede tornem-se cientes das mudanças ocorridas na topologia de maneira rápida e eficiente. Em modelos

centralizados e com pouco dinamismo, pode ser usada uma estratégia determinística para disseminação de eventos. Para as redes dinâmicas, a descoberta e disseminação de novidades aos participantes necessita de estratégias também dinâmicas e descentralizadas, que se adaptam a alterações imprevistas.

Uma abordagem promissora para tratar esse problema origina-se da observação da natureza. Certos grupos de seres vivos, tais como as colônias de formigas e enxames de abelhas, apresentam a chamada inteligência coletiva [Bonabeau et al. 1999]. Inteligência coletiva é a propriedade apresentada por sistemas constituídos por agentes dotados de comportamento autônomo, capazes de executar ações simples. Tais agentes, coletivamente, podem apresentar comportamento inteligente [White and Pagurek 1998]. Em outras palavras, as interações entre os diversos elementos é a chave para a realização da tarefa desejada.

A estratégia apresentada neste trabalho é baseada no algoritmo *EvAnt*, que utiliza inteligência coletiva para disseminar eventos em redes dinâmicas e descentralizadas. O algoritmo de inteligência coletiva é requisitado somente em situações nas quais há novidades a serem informadas aos elementos da rede. Estudos empíricos foram realizados a fim de avaliar a eficiência e eficácia da abordagem. A comparação da estratégia proposta foi realizada com a inundação (*flooding*) [Mullender 1993] e a disseminação epidêmica (*gossip*) [Eugster et al. 2004]. Dentre os aspectos avaliados encontram-se a velocidade que o algoritmo consegue disseminar novidades e a carga imposta à rede durante tal processo.

O restante deste trabalho está organizado conforme descrito a seguir. A seção 2 apresenta a estratégia proposta, sendo seguida da descrição das simulações e avaliação de resultados na seção 3. Na seção 4 são descritos os trabalhos relacionados. Por fim, as conclusões seguem na seção 5.

2. *EvAnt*

A solução proposta possui como base o algoritmo *EvAnt*, elaborado para este trabalho. O algoritmo faz uso de inteligência coletiva para disseminar, através de agentes móveis, novidades em redes dinâmicas e descentralizadas. Tais redes possuem comportamento imprevisível e podem sofrer alterações (a saída e entrada de nodos e links) frequentes.

2.1. Disseminação de novidades baseada em eventos

O algoritmo é disparado com a ocorrência de um evento. Denomina-se *evento*, ou *novidade*, alguma alteração no estado de um componente da rede. Uma alteração corresponde à criação ou eliminação de um enlace bem como o surgimento ou a saída de algum nodo da rede. Após a ocorrência de um evento, os nodos que o *detectaram* são aqueles que iniciam a *disseminação* das informações. Os nodos que inicialmente tornam-se cientes dos eventos e disparam a disseminação diferem de acordo com o tipo de evento:

surgimento de um nodo: a disseminação da novidade inicia-se pelo próprio nodo que acabou de surgir, bem como pelos nodos que são seus vizinhos;

saída de um nodo: os vizinhos do nodo que saiu da rede iniciam a disseminação;

criação de um enlace: os nodos que estão ligados pelo novo enlace iniciam a propagação do evento;

eliminação de um enlace: os nodos que estavam ligados pelo enlace eliminado iniciam a propagação da novidade.

2.2. Modelo do sistema

O ambiente em que o algoritmo é executado é uma rede dinâmica e descentralizada, como são as redes *peer-to-peer* não estruturadas e redes *ad hoc*. A entrada e saída de *hosts* pode ser intensa, de modo que as mudanças no estado da rede são frequentes e imprevisíveis. Outro ponto a ser observado é a total ausência de componentes que centralizam a tarefa de coletar e disseminar as novidades.

O modelo de sistema a ser considerado no algoritmo *EvAnt* é definido como um grafo não direcionado $G(t) = (V_t, E_t)$ [Nassu et al. 2007]. Tal grafo é composto, em um instante de tempo t , de um conjunto de vértices V_t e um conjunto de arestas E_t . Por ser definido em função do tempo, pode ocorrer que, para dois instantes de tempo físico distintos t_1 e t_2 , $G(t_1) \neq G(t_2)$. As seguintes características definem o modelo de sistema considerado:

1. Os nodos sempre possuem, para um dado instante de tempo t , o conhecimento correto sobre o estado dos vizinhos, isto é, dos vértices aos quais estão diretamente conectados através de um único enlace.
2. O envio de mensagens é livre de erros. O tempo para o envio de informações é não desprezível e não há limite de tempo definido para que uma mensagem chegue ao destino.
3. Não há relógio global compartilhado pelos nodos. Cada nodo possui apenas um relógio local e os diversos relógios locais dos nodos não são sincronizados entre si. Logo, a diferença entre dois instantes de tempo físico t e $(t + 1)$ não é a mesma para os vários participantes da rede.
4. O grafo que representa o sistema é não direcionado. Os enlaces são bidirecionais, ou seja, vale a igualdade $e_{ijt} = e_{jit}$.

2.3. Disseminação das formigas

As informações são disseminadas pela rede através de formigas, que são abstrações para agentes móveis. Um exemplo de como as mensagens são transmitidas entre os nodos é mostrado na Figura 1. Na primeira figura, 1(a), a formiga localiza-se no nodo 0, mostrado em cor cinza. Neste caso, a formiga carrega informações relativas aos enlaces que o nodo 0 possui, conforme mostrado no quadro à esquerda do nodo. O destino escolhido pela formiga é o nodo 1, caminho mostrado pelo enlace pontilhado. Na segunda figura, 1(b), o nodo 1 já foi atualizado com as informações trazidas pela formiga.

A disseminação das novidades, com o objetivo de permitir que as formigas consigam transmitir as informações para toda a rede, utiliza a abordagem baseada na estigmergia. Cada nodo contém um depósito de feromônios associado a cada um dos enlaces que possui. Quando uma formiga viaja de um nodo i em direção a algum vizinho j , a concentração $C_{ij}(t)$ de feromônios daquele enlace é incrementada em uma unidade, conforme mostra a expressão $C_{ij}(t) = C_{ij}(t - 1) + 1$ [Nassu et al. 2007].

Os feromônios também são utilizados para auxiliar na escolha do próximo destino das formigas. Essa escolha é feita de forma probabilística: enlaces com menor concentração de feromônios possuem maior probabilidade de serem escolhidos. A expressão $P_{ij}(t) = \frac{(C_{ij}(t)+1)^{-\alpha}}{\sum_{k=1}^{|e_{it}|} (C_{iE_{it}(k)}(t)+1)^{-\alpha}}$ define a probabilidade de uma formiga migrar do nodo i para o j no instante de tempo t . O tamanho da vizinhança do nodo i é representado

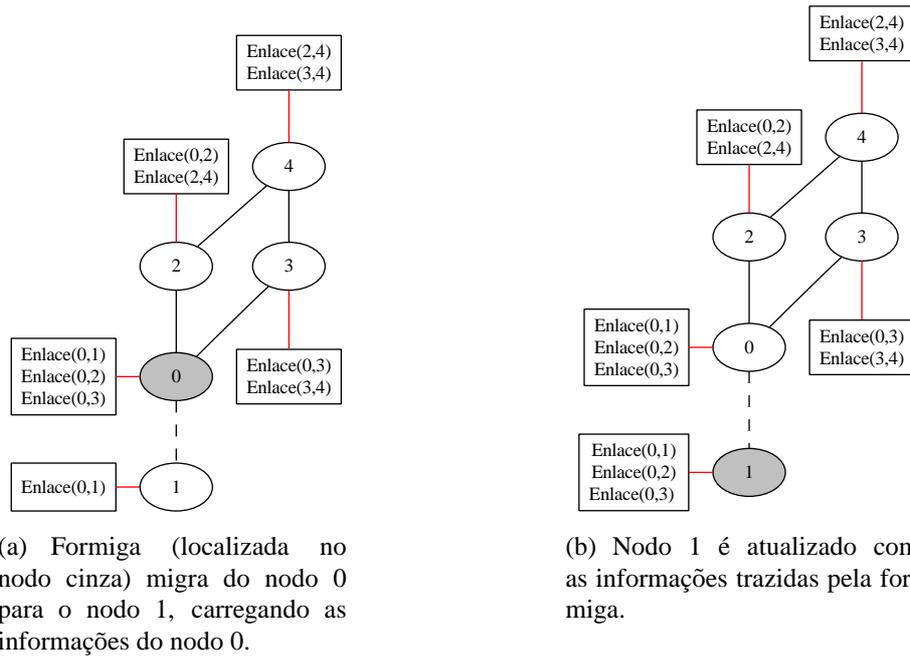


Figura 1. Disseminação de informações pela formiga.

por $|e_{it}|$ e $E_{it}(k)$ denota o k -ésimo vizinho do nodo i no tempo t . A constante α , tal que $\alpha > 0$, representa a força dos feromônios. Maior o valor de α , maior a diferença entre as probabilidades obtidas. Os feromônios, assim, servem para “afastar” as formigas umas das outras, fazendo com que elas sejam melhor distribuídas pela rede.

Com o passar do tempo, os feromônios enfraquecem. O enfraquecimento dos feromônios é dado pela expressão $C_{ij}(t) = C_{ij}(t_0) \cdot (1 - \rho)^{(t-t_0)}$. A nova concentração $C_{ij}(t)$ de feromônios do enlace e_{ij} no tempo t é dada em função da concentração do enlace em um instante de tempo anterior t_0 e uma constante ρ , tal que $0 \leq \rho < 1$. Essa constante controla a velocidade da evaporação. Maior o ρ , mais rapidamente os feromônios evaporam.

2.4. Controle da população de formigas

A quantidade de formigas na rede possui grande impacto no desempenho do algoritmo. O excesso de agentes faz com que as mesmas informações sejam repetidamente disseminadas pela rede. Por outro lado, com um número insuficiente de formigas as novidades não conseguem alcançar todos os nodos. Para controlar a população de formigas de maneira dinâmica, também são utilizados feromônios.

A concentração total $C_i(t)$ de feromônios de um nodo i é dada pela soma dos valores de feromônio de cada enlace desse nó, conforme mostrado na expressão $C_i(t) = \sum_{k=1}^{|e_{it}|} C_{iE_{it}(k)}(t)$. Esse valor é então comparado aos limites mínimo ($L_{min}(t)$) e máximo ($L_{max}(t)$), definidos pelas expressões $L_{min}(t) = \gamma_{min} \cdot |e_{it}|^\delta$ e $L_{max}(t) = \gamma_{max} \cdot |e_{it}|^\delta$, respectivamente. A constante δ , tal que $\delta \geq 0$, define o papel do tamanho da vizinhança sobre a definição dos limites. À medida que o valor de δ aumenta, maior a influência que o tamanho da vizinhança terá sobre os valores de $L_{min}(t)$ e $L_{max}(t)$. As constantes γ_{min} e γ_{max} devem ser escolhidas de modo que $\gamma_{min} > 0$ e $\gamma_{max} > \gamma_{min}$.

Caso $C_i(t)$ seja menor ou igual a $L_{min}(t)$, uma formiga é criada pelo nodo. Essa formiga carregará inicialmente as mesmas informações que aquele nodo possui. Se, por outro lado, $C_i(t)$ for superior ou igual a $L_{max}(t)$, todas as formigas localizadas no nodo são destruídas.

2.5. Especificação do algoritmo

A estratégia proposta é composta de duas partes principais: a primeira, apresentada no algoritmo 1 (*EvAnt*: ciclo do nodo), descreve o comportamento de um nodo i . O algoritmo 2 (*EvAnt*: ciclo da formiga), por sua vez, mostra as ações executadas por uma formiga f .

Inicialmente, no algoritmo 1, um novo evento faz com que os nodos que detectaram o acontecimento criem formigas para disseminar a novidade pela rede. Os nodos que detectaram o evento criam formigas em toda rodada na qual a concentração local de feromônios $C_i(t)$ for menor ou igual ao limite mínimo $L_{min}(t)$. Nodos que receberem formigas com novidades também criam um novo agente a cada rodada, enquanto a condição $C_i(t) \leq L_{min}(t)$ for respeitada. Caso, em alguma rodada, o nodo i possua a concentração local de feromônios $C_i(t)$ maior ou igual ao limite superior $L_{max}(t)$, todas as formigas que no momento estão no nodo i são destruídas.

O ciclo das formigas, mostrado no algoritmo 2, é composto por 5 ações. Inicialmente, a formiga, vinda do nodo i , chega ao nodo j e incrementa a concentração de feromônios relativa ao enlace (i, j) . Em seguida, a formiga atualiza o nodo com as novidades que carrega consigo. Após a formiga ficar inativa no nodo por um determinado período de tempo, que é parâmetro do algoritmo, a formiga então se atualiza com informações que ela não possui. Por fim, a formiga seleciona o próximo destino, de acordo com a expressão mostrada na linha 6 do algoritmo.

Algoritmo 1: EvAnt: ciclo do nodo i

```

1 while true do
2   |   evapora feromônios  $\rightarrow C_{ij}(t) = C_{ij}(t_0) \cdot (1 - \rho)^{(t-t_0)}$ 
3   |   if  $i$  detectou evento or ( $i$  recebeu formiga que carrega novas informações and  $C_i(t) \leq L_{min}(t)$ ) then
4   |   |   cria nova formiga
5   |   |   |   escala ciclo para a nova formiga (Algoritmo 2)
6   |   |   end
7   |   |   else
8   |   |   |   if  $C_i(t) \geq L_{max}(t)$  then
9   |   |   |   |   destrói todas as formigas localizadas no nodo  $i$ 
10  |   |   |   end
11  |   |   end
12 end
```

Algoritmo 2: EvAnt: ciclo da formiga f

```

1 while true do
2   |   formiga chega ao nodo  $j$ , vinda do nodo  $i \rightarrow C_{ij}(t) = C_{ij}(t-1) + 1$ 
3   |   formiga atualiza informações do nodo
4   |   formiga fica inativa no nodo por um determinado tempo
5   |   formiga atualiza-se com informações do nodo
6   |   formiga seleciona o próximo destino  $\rightarrow P_{ij}(t) = \frac{(C_{ij}(t)+1)^{-\alpha}}{\sum_{k=1}^{|E_{it}|} (C_{iE_{it}(k)}(t)+1)^{-\alpha}}$ 
7 end
```

3. Resultados das simulações

Ao longo desta seção serão apresentados resultados de estudos empíricos realizados para validar e avaliar o algoritmo proposto. Será mostrada uma comparação de resultados obtidos através da simulação entre o algoritmo *EvAnt*, a inundação (*flooding*) [Mullender 1993] e a disseminação epidêmica (*gossip*) [Eugster et al. 2004]. Os três algoritmos são utilizados na disseminação de informações sobre um evento detectado. Ambos os algoritmos foram escolhidos para a comparação por serem métodos de referência em disseminação de informações. Os três métodos serão avaliados através de simulações com o objetivo de observar a eficiência e eficácia dos algoritmos na atualização das informações da rede baseada na detecção de eventos.

3.1. Descrição das simulações

As simulações foram conduzidas conforme descrito a seguir. Inicialmente, todos os nodos da rede possuem, em suas descrições locais, informações corretas sobre o estado de todos os nodos da rede. Então, a partir desse estado, eventos são disparados ao longo do tempo e informações relativas ao processo da detecção desses eventos são coletadas. As métricas escolhidas possuem como objetivo avaliar a velocidade de detecção dos acontecimentos, bem como o peso que cada método impõe à rede para todos os nodos ficarem cientes dos eventos.

Para avaliar a velocidade da detecção de eventos, foram utilizadas duas métricas: latência (*latency*) e a taxa de ciência (*awareness rate*). A latência é definida como o número de rodadas necessário para que todos os eventos sejam detectados. A taxa de ciência tem como função mostrar a quantidade de nodos cientes dos eventos ao longo do tempo. Esta taxa pode ser obtida através da expressão $\frac{\sum_{i=1}^{|\varepsilon|} |\mu_i|}{|\varepsilon|}$, onde μ_i denota o número de nodos cientes do i -ésimo evento e $|\varepsilon|$ representa o tamanho do conjunto de eventos. Por exemplo: para uma rede com 100 nodos na qual foram disparados 3 eventos ($|\varepsilon| = 3$), a taxa de ciência é dada por $\frac{(|\mu_1|+|\mu_2|+|\mu_3|)}{3}$. Caso não haja eventos ($|\varepsilon| = 0$), a métrica não se aplica.

As 2 métricas utilizadas para medir a carga imposta pelo método de detecção de eventos são o número de mensagens trocadas (*# messages exchanged*) e a quantidade de mensagens criadas (*# agents created*). O número de mensagens trocadas representa a quantidade de formigas que se deslocaram de um nodo a outro, mais uma vez considerando um determinado intervalo de tempo. As mensagens criadas denotam o número de novas formigas criadas ao longo do processo de detecção de eventos.

Redes de topologia *powerlaw* [Bu and Towsley 2002] e aleatória foram utilizadas. Uma rede de topologia *powerlaw* apresenta um determinado conjunto de nodos fortemente conectado. Os nodos restantes são ligados a algum membro desse conjunto. Essa distribuição de nodos e enlaces é semelhante à encontrada na Internet. Uma rede de topologia aleatória possui os enlaces escolhidos de maneira randômica, sendo conexa e cada nodo com ao menos um enlace. As redes aleatórias possuem uma distribuição mais uniforme dos graus de seus nodos, se comparadas às redes *powerlaw*. Foram utilizadas redes de 1000, 2000 e 2500 nodos. Para cada tamanho, 20 redes foram geradas, 10 com topologia *powerlaw* e outras 10 com topologia aleatória. As redes *powerlaw* foram criadas de acordo com o seguinte critério [Bu and Towsley 2002]: $m_0 = 20\%$ do tamanho da rede e

sementes que variam de 0 a 9. As redes de topologia aleatória possuem o mesmo número de enlaces que as redes *powerlaw*.

As simulações tiveram tempo limite de 50 unidades de tempo. As rodadas, definidas como o intervalo no qual as métricas são coletadas, possuem duração de 0.2 unidades de tempo. Para cada uma das redes, um conjunto aleatório de eventos correspondente a 1% do tamanho da rede foi gerado. Esse conjunto de eventos foi estabelecido de modo que, após a sua ocorrência, a rede continue conexa. Os eventos possíveis são a criação e remoção de nodos e enlaces. Cada evento é disparado em intervalos de 5 rodadas (1 unidade de tempo).

A simulação do algoritmo *EvAnt* termina quando todos os eventos forem detectados (*awareness rate* igual ao tamanho da rede) ou o tempo limite da simulação for atingido. O algoritmo *EvAnt* foi comparado com a inundação (*flooding*) e a disseminação epidêmica (*gossip*).

No *flooding* [Mullender 1993], a disseminação de informações inicia-se a partir do ponto de origem dos eventos. Quando um nodo receber uma mensagem que traga novidades às informações contidas naquele nodo, essas informações locais são atualizadas bem como a mensagem de modo que ambos possuam as mesmas informações. Essa nova mensagem é então disseminada por inundação a partir do nodo. Os termos mensagem, agente e formiga referem-se à mesma entidade, responsável por disseminar os eventos pela rede. O algoritmo termina após uma rodada na qual nenhum nodo disseminou novas mensagens.

No *gossip* [Eugster et al. 2004], a disseminação de informações também é iniciada no ponto onde os eventos são originados. Os nodos que recebem mensagens contendo novas informações são atualizados com as novidades e atualizam a mensagem de modo que ambos contenham as mesmas informações. O nodo então dissemina essa mensagem para $f \geq 1$ vizinhos escolhidos aleatoriamente. Tal mensagem é disseminada pelo nodo durante $t \geq 1$ rodadas. Os parâmetros t e f são os mesmos para todos os nodos da rede.

O *gossip* utilizado neste artigo possui como parâmetros $f = \log_{10}(N)$ e $t = 1$. Esse valor foi escolhido de acordo com o critério explicado em [Bollobas 2001]. Os parâmetros utilizados para o algoritmo *EvAnt* são mostrados à seguir:

Força dos feromônios (α): 8

Taxa de evaporação dos feromônios (ρ): 0.4

Peso da vizinhança (δ): 0.5

Limite inferior para a concentração local de feromônios (γ_{min}): $\frac{\log(N)/4}{|vizinhança\ média|^\delta}$

Limite superior para a concentração local de feromônios (γ_{max}): $100 \times \gamma_{min}$

Tempo de inatividade das formigas: 0.2 segundos (1 rodada).

Para obter uma comparação com o *gossip*, foi estipulado que o *EvAnt* deve disseminar mensagens por, em média, $\log_{10}(N)$ rodadas. Assim, o limite mínimo $L_{min}(t)$ da concentração de feromônios foi fixado como $\log_{10}(N)$, então, o valor da constante γ_{min} foi obtido. Com o valor de γ_{min} fixado, os demais valores foram obtidos através de análise experimental, ou seja, diversos valores foram simulados até que a melhor combinação de constantes fosse obtida.

Pode-se obter um paralelo sobre a maneira como o *gossip* e o *EvAnt* disseminam as mensagens. No *gossip*, cada nodo dissemina $\log_{10}(N)$ mensagens, todas na mesma

rodada. Por outro lado, no algoritmo *EvAnt* cada nodo dissemina, em média, $\log_{10}(N)$ mensagens, uma por rodada.

3.2. Resultados obtidos

Os resultados das simulações são mostrados a seguir. Na Figura 2 temos os gráficos referentes às médias obtidas das 10 redes *powerlaw* de 2500 nodos. A Figura 2(a) apresenta a latência obtida por cada estratégia. Nas Figuras 2(b) e 2(c) podemos observar a o número de mensagens trocadas e a quantidade de formigas criadas ao longo do tempo, respectivamente. De acordo com a Figura 2(a), o *gossip* não foi capaz de deixar todos os nodos cientes das novidades. Os outros dois algoritmos conseguiram comunicar todos os eventos, com o *flooding* sendo, em média, 18.2% (28 rodadas) mais rápido que o *EvAnt*. Para tal, o *flooding* trocou aproximadamente 19.7% (ou 42000) mensagens a mais do que o *EvAnt*, conforme pode ser visto em 2(b). A diferença entre a quantidade de mensagens criadas ao longo das simulações favorece o *EvAnt*, pois ele não necessita criar cópias das mensagens que dissemina aos vizinhos. Cerca de 99.4% agentes a mais foram criados pelo *flooding* (Figura 2(c)). O *gossip*, mesmo não conseguindo atingir todos os nodos da rede, criou cerca de 96000 mensagens a mais que o algoritmo *EvAnt*, uma diferença de 98.65%. A Figura 2(d) mostra a *awareness rate*, ou seja, reflete com mais detalhes os resultados mostrados em 2(a).

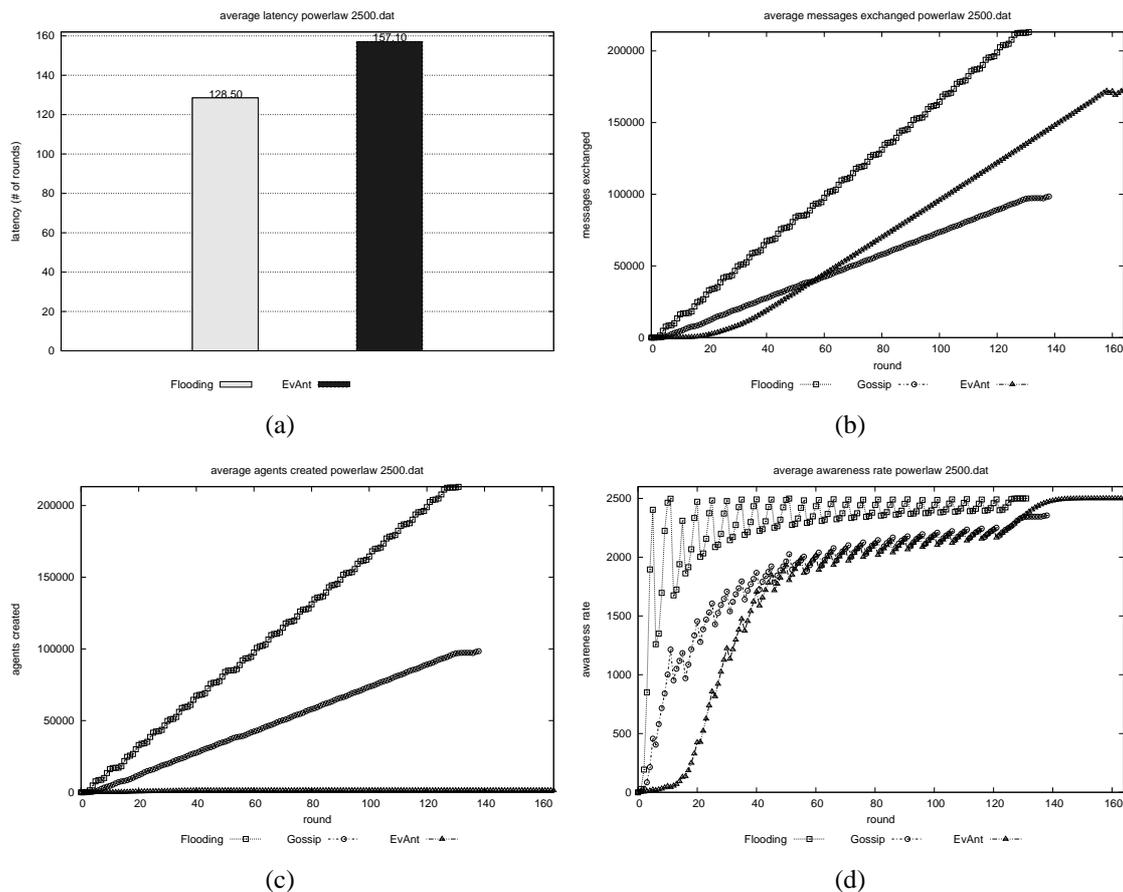


Figura 2. Gráficos referentes às redes *powerlaw* de 2500 nodos.

Os gráficos referentes às redes de topologia aleatória de 2500 nodos, bem como

os resultados referentes às redes de 1000 e 2000 nodos são omitidos aqui por restrições de espaço. Os resultados obtidos em tais redes foram semelhantes aos apresentados nas redes *powerlaw* de 2500 nodos.

Um resumo dos resultados referentes às redes de 2500 nodos é mostrado na Tabela 1. Para cada métrica, o valor mostrado na parte superior refere-se à média dos dados obtidos nas simulações. Os valores mostrados na parte inferior dizem respeito aos desvios padrão obtidos.

Para as duas topologias, *powerlaw* e aleatória, pode ser observado que tanto no *flooding* quanto no *EvAnt* toda a rede consegue ficar ciente de todas as novidades. O *flooding*, como um algoritmo de disseminação confiável, sempre consegue tornar a rede ciente de todos os acontecimentos. Uma vez que a *awareness rate* do algoritmo *EvAnt* é a mesma obtida pelo *flooding*, conclui-se que o algoritmo proposto consegue também deixar toda a rede ciente dos eventos. O *gossip* em nenhum caso conseguiu disseminar os eventos para todos os nodos. A latência obtida pelo *flooding*, como era de se esperar, é menor que a obtida pelo *EvAnt*, em todos os casos.

Ao se analisar o número de mensagens trocadas, percebe-se uma diferença considerável entre os métodos. O algoritmo *EvAnt* sempre consegue disseminar todas as informações pela rede com um menor número de mensagens que o *flooding*, essa diferença chegando a ser de mais de 29%, conforme pode ser observado nas redes aleatórias de 2500 nodos. Em alguns casos, como para as redes *powerlaw* de 2500 nodos, o *gossip* troca menos mensagens que as outras duas estratégias. Porém este algoritmo não atingiu o objetivo de disseminar as informações para todos os nodos, como explicado acima. Ainda, em outras situações, como nas redes de topologia aleatória de 2500 nodos, o *gossip*, mesmo não sendo capaz de tornar toda a rede ciente das novidades, trocou mais mensagens do que o algoritmo *EvAnt*.

Esse é um forte indício da eficiência do algoritmo *EvAnt*, uma vez que ele é capaz de alcançar a mesma taxa de ciência do *flooding* com uma menor quantidade de mensagens disseminadas ao longo do processo.

Por fim, quanto ao número de mensagens criadas ao longo das simulações, há novamente um ganho significativo por parte do *EvAnt*. Isso ocorre pois, a cada vez que o *flooding* ou o *gossip* recebem uma mensagem que carrega novidades, eles necessitam criar cópias dessa mensagem para então espalhá-las pela rede. No algoritmo *EvAnt*, é criada apenas uma formiga por nodo, a cada rodada na qual a condição $C_i(t) \leq L_{min}(t)$ for atendida.

	<i>powerlaw</i>			topologia aleatória		
	<i>flooding</i>	<i>gossip</i>	<i>EvAnt</i>	<i>flooding</i>	<i>gossip</i>	<i>EvAnt</i>
latência	128.5 0.527	– –	157.1 3.178	131.1 0.737	– –	161.2 1.98
mensagens trocadas	212591.8 2351.7	97304.1 1942	170525.2 4232.6	211328.4 2377.9	185029 1516.7	148998.3 2820.335
mensagens criadas	212591.8 2351.7	97304.1 1942	1311.2 25.7	211328.4 2377.9	185029 1516.7	1098 15.8
<i>awareness rate</i>	2500 0	2344.9 10.5	2500 0	2500 0	2497 0.44	2500 0

Tabela 1. Resultados para as redes de 2500 nodos.

A fim de verificar estatisticamente se os resultados podem ser considerados re-

levantes, foi utilizado o *teste de Wilcoxon* [Wilcoxon 1945]. Esse teste é utilizado para verificar se um determinado conjunto de dados apresenta resultados significativamente diferentes em relação a outro. A comparação foi realizada apenas entre os algoritmos *EvAnt* e *flooding*, uma vez que, conforme já mostrado, o *gossip* não consegue tornar toda a rede ciente dos eventos. O teste de Wilcoxon confirmou que, para todas as métricas estudadas, com exceção da *awareness rate*, igual para ambos os métodos, os conjuntos de dados obtidos pelos algoritmos *EvAnt* e *flooding* são mesmo distintos.

Através da análise conjunta dos resultados do teste de Wilcoxon com a Figura 1, pode-se observar um claro caso de escolha conflitante (*trade-off*) entre velocidade e carga imposta à rede. De um lado, o *flooding* oferece rápida detecção de eventos porém com um alto número de mensagens trocadas e criadas ao longo do processo. Do outro lado há o algoritmo *EvAnt*, que, embora demore algumas rodadas a mais para tornar toda a rede ciente das novidades, realiza essa tarefa com uma menor quantidade de mensagens trocadas e com uma diferença muito expressiva no número de mensagens criadas.

4. Trabalhos relacionados

As duas estratégias utilizadas para comparação com a solução proposta são consideradas referência para a disseminação de eventos: a inundação (*flooding*) [Mullender 1993], que garante que as informações chegarão a todos os nodos da rede, e a disseminação epidêmica (*gossip*) [Eugster et al. 2004], que é uma alternativa menos custosa que o *flooding*, mas é probabilística, não garantindo que todos os nodos serão alcançados.

Alguns métodos têm utilizado inteligência coletiva para realizar determinadas tarefas em redes *ad hoc* e *peer-to-peer*. Aqui são citados aqueles que possuem maior relação com a estratégia proposta neste trabalho. Nenhum desses métodos é diretamente comparável pelas diferenças de asserções e/ou objetivos. Em [Brocco et al. 2009], com o objetivo de localizar recursos em *grid*, a disseminação de informações ocorre através de agentes móveis (formigas) especializadas em várias funções, tais como a manutenção de enlaces em situações de baixo tráfego. Um ponto importante a ser destacado neste trabalho é que ele também cria formigas quando o estado dos enlaces dos nodos é alterado. O algoritmo altera a topologia da rede, isto é, cria e remove enlaces, de modo a manter uma rede fracamente conexa.

O trabalho [Ahmed and Far 2007] também trata sobre disseminação de agentes em redes dinâmicas, porém sem utilizar a abordagem da Otimização por Colônia de Formigas. Segundo os autores, as estratégias baseadas em formigas até então apresentadas continuavam a disseminar agentes mesmo em situações nas quais não haviam novidades a serem disseminadas. Os agentes são hierarquizados em dois tipos. Um tipo de agente percorre a rede em busca de informações e as informa ao outro tipo. Há uma diferença conceitual neste trabalho, uma vez que existe a hierarquização dos nodos em dois tipos.

A disseminação de informações através de formigas é utilizada em [Aissani et al. 2007] para auxiliar no roteamento. Tabelas cache em cada nodo colaboram para as mensagens enviadas encontrarem o destino. Este trabalho também possui formigas especializadas em realizar determinadas tarefas. A circulação de pacotes na rede é permanente, com o objetivo de manter as tabelas cache dos nodos atualizadas.

O roteamento também é tratado em [Gopalan et al. 2006] por meio de agentes móveis, especificamente utilizando Otimização por Colônia de Formigas. Porém, este

trabalho possui um componente centralizador, uma vez que um determinado nodo é que concentra as informações sobre as rotas da rede.

Em [Wei et al. 2009] é proposto um protocolo para a descoberta da topologia em redes *ad hoc* através do uso de agentes móveis. Uma taxa de mudança dos vizinhos de um nodo foi estabelecida. Quanto maior for essa taxa, maior chance que os agentes terão de migrarem para aquele nodo. Por outro lado, nodos cuja vizinhança pouco muda ou muito visitados destroem os agentes que ali chegam.

O trabalho [Nassu et al. 2007] trata sobre a descoberta e disseminação da topologia de redes dinâmicas e descentralizadas através de agentes móveis, utilizando para isso inteligência coletiva, mais especificamente a Otimização por Colônia de Formigas. Informações contidas tanto pelos nodos quanto pelos agentes, após passarem de uma determinada idade, são descartadas. O algoritmo também dissemina formigas continuamente pela rede, em contraste com a proposta de disseminação de informações baseada em eventos. Além disso, os relógios locais de cada nodo são sincronizados entre si, de modo que a diferença entre dois instantes de tempo físico é a mesma para todos os nodos. Os conceitos de estigmergia e método de disseminação de informações apresentados pelo algoritmo *EvAnt* foram adaptados deste trabalho. Uma comparação direta dos dois métodos não foi possível pois cada estratégia baseia-se em um modelo de sistema diferente.

5. Conclusão

Este artigo apresentou o algoritmo *EvAnt*, uma estratégia para a disseminação de informações em redes dinâmicas e descentralizadas. Através de formigas que se deslocam pela rede, as informações são repassadas aos nodos. Cada nodo possui um depósito de feromônios relativo a cada enlace. Quanto maior a concentração de feromônios desse depósito, maior a frequência com que o enlace foi utilizado. A escolha dos destinos das formigas é tomada com base apenas em informações locais ao nodo no qual ela se localiza. Para que os agentes sejam distribuídos de maneira adequada na rede, os enlaces menos utilizados possuem maior probabilidade de serem escolhidos. Os depósitos de feromônios também são utilizados para controlar a população de formigas, de modo que não haja excesso ou falta de agentes.

Com o objetivo de não gastar recursos desnecessários da rede, a movimentação de agentes ocorre apenas quando há novidades a serem disseminadas. A estratégia mostrou-se eficiente e eficaz na tarefa à qual se propõe, uma vez que os experimentos mostraram que o algoritmo *EvAnt* é capaz de deixar toda a rede ciente dos eventos com uma menor quantidade de mensagens do que os métodos com o qual foi comparado.

Trabalhos futuros incluem considerar alterações em determinados pontos do modelo de sistema apresentado, como por exemplo considerar uma rede com enlaces direcionados. Uma implementação do algoritmo *EvAnt* para redes *peer-to-peer* deverá ser desenvolvida.

6. Agradecimentos

Este trabalho foi possível graças ao apoio do CNPq, projetos 304013/2009-9 e 303761/2009-1. O trabalho foi parcialmente financiado pela FINEP através dos projetos CT-INFRA/UFPR.

Referências

- Ahmed, A. and Far, B. (2007). Performance of mobile agent based network topology discovery. In *Electrical and Computer Engineering, 2007. CCECE 2007. Canadian Conference on*, pages 66–69.
- Aissani, M., Fenouche, M., Sadour, H., and Mellouk, A. (2007). Ant-dsr: Cache maintenance based routing protocol for mobile ad-hoc networks. In *Telecommunications, 2007. AICT 2007. The Third Advanced International Conference on*, pages 35–35.
- Bollobas, B. (2001). *Random graphs*. Cambridge Univ. Press.
- Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. Oxford University Press, USA.
- Brocco, A., Malatras, A., and Hirsbrunner, B. (2009). Proactive information caching for efficient resource discovery in a self-structured grid. In *BADS '09: Proceedings of the 2009 workshop on Bio-inspired algorithms for distributed systems*, pages 11–18, New York, NY, USA. ACM.
- Bu, T. and Towsley, D. (2002). On distinguishing between Internet power law topology generators. In *IEEE INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 2.
- Dong, C., rong Cheng, X., and quan Zhang, M. (2006). Research of mobile agent based network topology discovery. In *Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on*, volume 1, pages 733–736.
- Eugster, P., Guerraoui, R., Kermarrec, A., and Massoulié, L. (2004). Epidemic information dissemination in distributed systems. *Computer*, 37(5):60–67.
- Gopalan, N., Mala, C., Shriram, R., and Agarwal, S. (2006). Multicast tree computation for group communication in mobile networks using optimization techniques. In *Ad Hoc and Ubiquitous Computing, 2006. ISAUHC '06. International Symposium on*, pages 88–93.
- Mullender, S. (1993). *Distributed Systems*. Addison Wesley Publishing Company.
- Nassu, B., Nanya, T., and Duarte, E. (2007). Topology discovery in dynamic and decentralized networks with mobile agents and swarm intelligence. *Proceedings of 7th ISDA, IEEE Computer Society, Washington*, pages 685–690.
- Wei, J., Guo, W., Su, J., and Tang, W. (2009). Mobile agent based topology discovery in mobile ad hoc networks. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1–4.
- White, T. and Pagurek, B. (1998). Towards multi-swarm problem solving in networks. In *Proceedings of Third International Conference on Multi-Agent Systems (ICMAS'98)*, pages 333–340.
- Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.