

# Resultados da simulação de adoção de algoritmo bioinspirado para o roteamento de dados

## Eduardo Fávero Pacheco da Luz

Instituto Nacional de Pesquisas Espaciais, INPE  
Programa de Pós-Graduação em Computação Aplicada, CAP  
12227-010, São José dos Campos, SP  
E-mail: eduardofpl@hotmail.com

## **Raul Abreu de Assis**

Universidade do Estado de Mato Grosso, UNEMAT  
Campus Universitário de Barra do Bugres  
Departamento de Matemática  
78390-000, Barra do Bugres, MT  
E-mail: raulassis@yahoo.com

A Ciência da Computação pode se valer de exemplos encontrados na natureza, no momento em que se verifica nesta a última, a existência de soluções simples e elegantes para a resolução de problemas, principalmente problemas relacionados à otimização e reconhecimento de padrões. Estas soluções são encontradas nas mais variadas espécies de seres vivos que melhor se adaptaram às necessidades ao longo de milhares de anos de evolução e de eliminações de indivíduos mais fracos pela seleção natural. O resultado é visto, como já mencionado, na solução otimizada de problemas.

Para determinadas classes de problemas, as técnicas tradicionais de computação não se mostraram adequadas para encontrar a solução. E neste momento é necessário levar em conta o grande exemplo dado pela natureza através de seus sistemas naturais. A Ciência da Computação tem se aliado às Ciências Biológicas no momento de transcrever os comportamentos que levam à solução de problemas para equações matemáticas, que podem ser facilmente manipulados por técnicas implementadas em computadores digitais.

As técnicas utilizadas para o projeto de algoritmos computacionais baseados em princípios biológicos são conhecidas por Computação Biológica ou Biocomputação. Incluem-se nestas técnicas as redes neurais artificiais, algoritmos evolutivos, sistemas imunológicos, robótica adaptativa e evolutiva, computação baseada em moléculas e otimização baseada em colônias de formigas ou inteligência de enxames (Figura 1).

Para Bonabeau [2], a vida em sociedade possui algumas características que dentro de uma comunidade de insetos em muito se difere da sociedade humana. Mas, apesar dos indivíduos exibirem uma baixa capacidade de raciocínio, observa-se um alto grau de organização nas colônias. Questões como “Quem os governa? Como são divididas as tarefas?”, dentre outras, se tornam de difícil resposta.



Figura 1: Formiga argentina (*Linepithema humile*)

Segundo Assis [1], sob diversas circunstâncias, colônias de insetos sociais, em particular, colônias de formigas, devem tomar “decisões” para o bem geral da colônia. É interessante notar que essas decisões são tomadas sem qualquer destaque de líderes dentro da colônia.

Estas características comportamentais existentes em sociedades biológicas, principalmente em colônias de formigas, abelhas, vespas, cupins e lagartos, sempre foram fontes de inspiração para estudos na área de biologia. No momento em que as técnicas computacionais se aliaram a esses esforços, essas características receberam uma elevada importância quando aplicadas à resolução de problemas que afetam diretamente os interesses dos seres humanos. As características que mais chamaram a atenção devido a sua aplicabilidade são:

- ## Autonomia;
- ## Auto-organização;
- ## Comportamento emergente;
- ## Funcionamento distribuído.

Um dos principais experimentos que acabaram por comprovar as características de otimização existentes em insetos sociais foram promovidos por Deneubourg, visto em [2]. As formigas foram o alvo dessas experiências que ficaram conhecidas como as Experiências da Ponte Binária.

Nestas experiências, uma fonte de alimento é separada do ninho por uma ponte com dois caminhos de distâncias iguais (Figura 2), e através de escolhas aleatórias e/ou seleção do caminho por análise da trilha de feromônio já existente, ou não, a formiga seleciona entre os possíveis caminhos.

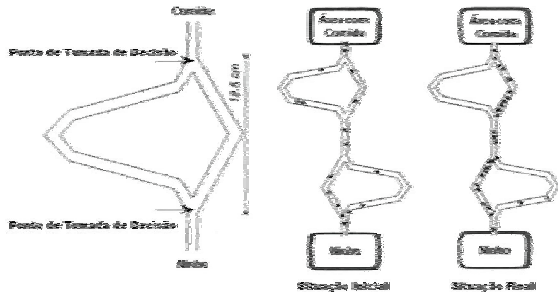


Figura 2: Modelo esquemático da ponte binária

A Biocomputação, através do estudo destes comportamentos de busca de presas de várias espécies de insetos, mas em principal as formigas, inspirou a criação de algoritmos de otimização, baseados na inteligência coletiva dessas espécies.

Esses estudos levaram à criação em particular de um algoritmo para roteamento de dados em redes de comunicação de larga escala, algoritmo esse conhecido como AntNet, que é descrito junto com outros algoritmos bioinspirados nas obras [2], [3] e [4].

O referido algoritmo foi desenvolvido por Gianni Di Caro e Marco Dorigo, tendo em vista a apresentação de um algoritmo adaptativo para roteamento de dados baseado em colônias de formigas. Seu principal objetivo é a construção de uma tabela de roteamento adaptada às condições de tráfego existentes em determinado instante.

O principal esquema de seleção do AntNet é representado pela Equação 1, que define a probabilidade  $P_{ij}^k(t)$  de uma rota qualquer a ser tomada no momento de tempo  $t$ , levando em consideração a quantidade de feromônio existente no trecho  $v_{ij}$ . Os parâmetros  $\zeta$  e  $\eta$  são reguláveis e determinam a influencia da concentração de feromônio e a visibilidade, respectivamente, no momento da seleção.  $C_{ij}(t)$  determina o custo associado à rota, ou seja, é variável dependente do custo da rota.

$$P_{ij}^k(t) = \frac{\tau_{ij}^k \cdot \eta^{\zeta} \cdot \Psi / C_{ij}(t)}{\sum_{i \in J_i^k} \tau_{ij}^k \cdot \eta^{\zeta} \cdot \Psi / C_{ij}(t)} \quad (1)$$

Informalmente, o AntNet pode ser descrito pelas diretrizes que seguem:

- ≠ Formigas *Forward*  $F_{s \downarrow d}$  são lançadas em intervalos de tempo regular de cada nó  $s$  na rede, com uma determinação de destino  $d$  aleatória. O destino é escolhido para se adaptar ao padrão de tráfego do momento;
- ≠ Cada formiga *Forward* seleciona o próximo nó através da rede, que ainda não foi visitado, seguindo uma tabela de probabilidades. A chance probabilística é definida proporcionalmente por fatores de acordo com o status da rede;
- ≠ A identificação de cada nó visitado e o tempo gasto de seu lançamento do nó anterior até sua chegada a este último é armazenada em um endereço de memória designado por  $S_{s \downarrow d}$  e é carregado pela formiga *Forward*;
- ≠ Se um ciclo é detectado, isto é, se uma formiga é forçada a retornar para um nó já visitado, porque um de seus vizinhos tinha um estado de “não-visitado”, as informações do endereço de memória correspondentes a este nó são destruídos;
- ≠ Quando a formiga  $F_{s \downarrow d}$  atinge o nó de destino  $d$ , ela cria a formiga  $B_{d \uparrow s}$ , transfere toda a informação de sua memória e morre;
- ≠ A formiga *Backward* faz o mesmo caminho de volta, ou seja, percorre os mesmos nós que estão armazenados nas informações transmitidas por sua antecessora, e no momento em que chega ao nó especificado, as informações com relação ao tempo gasto que a primeira formiga levou para atravessar determinado caminho são atualizados, e através de estatísticas, estes resultados são convertidos para parâmetros probabilísticos.

Após o levantamento bibliográfico referente às bases e ao funcionamento do algoritmo bioinspirado AntNet, assim como a verificação dos resultados obtidos pelos autores, foram executadas simulações com as informações obtidas, levando em consideração a composição nacional da principal rede de comunicação brasileira, o backbone da EMBRATEL.

O software utilizado para os estudos foi o OMNeT++, um sistema para simulação de eventos discretos modulares orientados a objeto. Seu nome advém de *Objective Modular Network Testbed in C++*, onde é facilmente verificável a grande influência da orientação a objeto existente na linguagem C++. Quando trabalhando em sua versão visual, é utilizado o editor GNED (Graphical NED Editor), onde NED é a linguagem de programação utilizada, muito integrada a C++, principalmente por suportar a possibilidade de descrição modular de uma rede [5].

Foi utilizada uma distribuição gratuita, regida pelos termos da licença geral de público uso (*General Public License – GPL*) de uma codificação do AntNet, disponibilizada pelo Professor Muddassar Farooq, do Departamento Informatik III da Universitaet Dortmund da Alemanha.

Com fins de comparação, também foi utilizada uma

distribuição da simulação do OSPF, o algoritmo de roteamento atualmente presente na Internet. Essa versão acompanha os arquivos de exemplo do OMNeT++.

As localizações dos nós das redes utilizadas para comparação foram alteradas, visando deixar-las similares à topologia do backbone da EMBRATEL, possibilitando assim uma estimativa dos resultados da utilização desses algoritmos (Figura 3).

Pelo fato da não disponibilização da totalidade dos dados necessários à reconstrução nos detalhes da malha da topologia, foram utilizadas larguras de banda padronizadas, assim como foram definidas métricas baseadas na escala de distância do espaço euclidiano. Melhores resultados são apresentados quando a métrica é baseada em saltos (*hops*) entre os pontos de conexão. Porém a situação pode se assemelhar a uma comparação com a distância no espaço, conforme variação da topologia.



Figura 3: Localizações dos centros de roteamento

A geração dos pacotes de dados, assim como dos agentes *Forward* e *Backward*, utilizados pelo AntNet, também obedeceram a uma distribuição uniforme, sem a presença de *hotspots* (pontos com alta concentração de geração de pacotes), caracterizando, assim, uma configuração de rede igualitária no seu amplo sentido.

Foram executadas dez (10) repetições das simulações, com regulamentação de mil e quinhentas (1.500) unidades de tempo simulado para execução de cada instância, tanto com roteamento através do AntNet, quanto do OSPF.

O tempo foi simulado discretamente, o que significa dizer que a cada ocorrência de evento interno da simulação (início de uma transmissão de pacote, fim da transmissão de um pacote, estouro de tempo de vida de um pacote, entre outros), ocorre

uma chamada interna nas funções do OMNeT++, armazenando informações relativas a cada uma dessas ocorrências, para posterior análise e conversão para um equivalente em tempo real [5].

Os resultados obtidos (Figura 4) são extremamente similares aos obtidos pelos autores do algoritmo, validando assim a utilização dessa nova possibilidade.

Quando analisado o tempo de atraso dos pacotes, removendo os cinco por cento (5%) dos extremos, visando o resultado não ser influenciado pelos pacotes extremamente rápidos ou lentos, temos uma diferença muito significativa de 5,316751 segundos (unidade de tempo da simulação) entre o algoritmo OSPF e o AntNet.

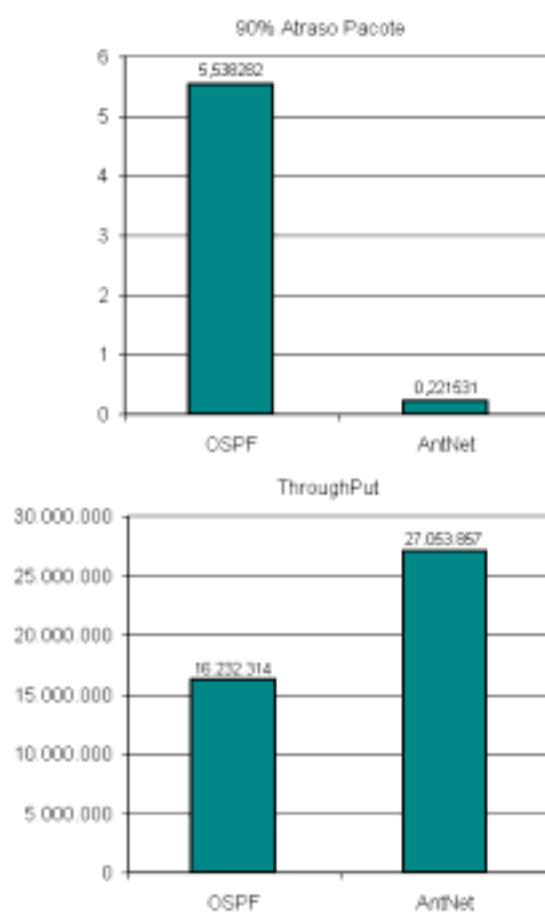


Figura 4: Resultados das simulações

A quantidade de pacotes que percorreram a malha do backbone também seguiu os mesmos padrões, uma diferença de 33,33% (trinta e três virgula trinta e três por cento) entre o OSPF (com 16.232.314 pacotes) e o AntNet (com 27.053.857 pacotes).

Ambos os resultados, favoravelmente, apontam o AntNet como um algoritmo de grandes vantagens sobre o OSPF.

As simulações computacionais utilizadas como ferramentas avaliativas desses aspectos qualitativos e quantitativos necessitam, agora, de estudos mais aprofundados, principalmente no que se refere à

interligação dos vários backbones que, interconectados através dos continentes, constituem a Internet propriamente dita.

A questão agora se centra na variabilidade dos diversos sistemas que são utilizados, abrindo assim, várias alternativas de estudos relacionados à adaptabilidade e aceitação por parte dos diversos mantenedores, no que se refere a adoção integral de algoritmos bioinspirados para o controle do roteamento de dados em redes de comunicação de larga escala.

## Referências

- [1] R. A. Assis. Modelos em Estratégias de Forrageamento de Formigas. Campinas: UNICAMP, 2003. 83 p. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, 2003.
- [2] E. Bonabeau, “Swarm Intelligent: From Natural do Artificial Systems”, Sante Fe Institute, New York, 1999.
- [3] G. Di Caro, AntNet: A Mobile Agents Approach to Adaptive Routing, *Technical Report IRIDIA*, 12 (1997).
- [4] G. Di Caro, AntNet: Distributed Stigmergetic Control for Communications Networks, *Journ. Artif. Intel. Res.*, 9 (1998) 317-365.
- [5] A. Varga. OMNeT++ User manual. Version 3.0 preview 1. 1,48 MB. Formato PDF. Disponível em: <<http://www.omnetpp.org>>. Acesso em: 03 fev. 2004.