

Grafos Evolutivos e Redes Tolerantes a Atrasos

Proposta de Monografia

15 de junho de 2009

Aluno: César Gamboa Machado - n°USP 5638870

Aluno: Paulo Floriano - n°USP 5639001

Supervisor: Prof. Dr. Alfredo Goldman vel Lejbman

Observação: A pesquisa sobre DTNs está sendo feita em um grupo de cinco alunos. Este grupo consiste dos alunos Adriano Tabarelli, Caio Cestari Silva, Cássia Garcia Ferreira, César Gamboa Machado e Paulo Henrique Floriano. Apesar de termos dividido os assuntos pesquisados para o desenvolvimento das monografias em grupos menores, todos tem uma base em comum.

1 Resumo da Monografia

Redes Tolerantes a Atrasos (em inglês, Delay Tolerant Networks, ou DTNs)[7] são redes que sofrem grandes mudanças de topologia e tem baixa conectividade. Em redes assim, desconexões e atrasos na entrega de mensagens são frequentes, fazendo com que protocolos de rede comuns como o TCP/IP se tornem ineficazes.

Exemplos de DTNs incluem redes móveis ad-hoc, redes de satélites e a planejada Internet Interplanetária.

Um problema comum em redes assim é como entregar mensagens entre dois pontos de forma eficiente.

Existem diversos algoritmos de roteamento para DTNs em que não há informações sobre conexões futuras. Exemplos de algoritmos assim são o roteamento epidêmico[9], o spray-and-wait[8], o PROPHET [5] e o Maxprop

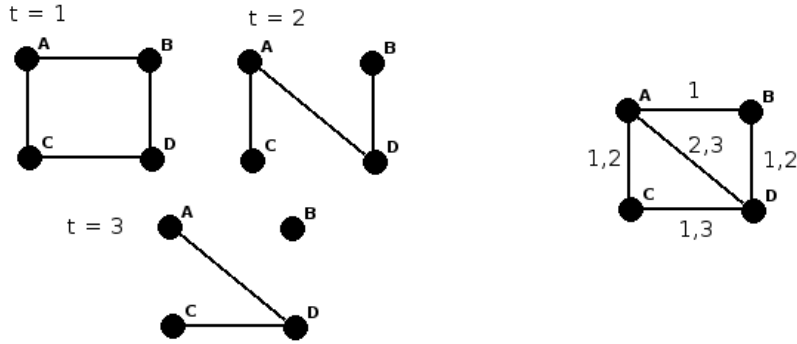
[1]. Tais algoritmos geralmente criam várias cópias da mesma mensagem na rede, de forma a aumentar a probabilidade de entrega, porém, existe uma possibilidade de sobrecarga da rede devido ao grande número de mensagens circulando.

Porém, se temos informações sobre todas as conexões que irão ocorrer na rede (como no caso da Internet Interplanetária, por exemplo), é possível encontrar rotas ótimas para as mensagens, diminuindo o número de pacotes perdidos e de cópias de uma mesma mensagem transitando na rede. Para isso, precisamos de uma estrutura que guarde tais informações.

Definição 1 (Grafos Evolutivos) *Sejam $G = (V_G, E_G)$ um grafo e $S_G = G_0, G_1, \dots, G_\tau$ ($\tau \in \mathbb{N}$) um conjunto ordenado de subgrafos de G tal que $\cup_{i=1}^\tau G_i = G$. O sistema $\mathcal{G} = (G, S_G)$ é chamado de Grafo Evolutivo.*

Grafos Evolutivos podem ser entendidos como grafos em que cada arco tem uma lista de intervalos de tempo na qual esta está ativo e são muito úteis para a representação de DTNs previsíveis.

A figura 1(b) mostra um exemplo de Grafo Evolutivo.



(a) Conexões em três instantes de tempo.

(b) Grafo evolutivo correspondente a 1(a)

Figura 1: Conexões e Grafo Evolutivo correspondente

Definição 2 (Jornada) *Seja $R = e_1, e_2, \dots, e_k$ ($e_i \in E_G$) um caminho em G , e $R_\sigma = \sigma_1, \sigma_2, \dots, \sigma_k$ ($\sigma_i \in [1, \tau]$, $\sigma_i \leq \sigma_j \forall i < j$) um agendamento indicando quando cada arco de R deve ser atravessado. Então dizemos $\mathcal{J} = (R, R_\sigma)$ é uma jornada em \mathcal{G} .*

Podemos definir três tipos de jornadas ótimas em um grafo evolutivo, levando em conta três medidas distintas:

- O instante de chegada da mensagem ($\sigma_{|R|}$). Chamamos de *Foremost Journey* a jornada que chega no nó destino mais cedo.
- O número de arcos percorridos ($|R|$). A jornada que percorre o menor número possível de arcos entre a origem e o destino é chamada *Shortest Journey*.
- O tempo de percurso, ou seja, a diferença entre o instante de envio (σ_1) e o instante de chegada ($\sigma_{|R|}$). A jornada com menor tempo de percurso é chamada *Fastest Journey*.

Algoritmos para calcular essas três jornadas foram propostos em [10]. No entanto, o algoritmo *Fastest Journey* não foi implementado ou usado em contextos reais.

2 Objetivos

O objetivo deste trabalho é o estudo dos algoritmos conhecidos para encontrar jornadas em Grafos Evolutivos, e a comparação destes com os algoritmos para entrega de mensagens em DTNs onde a rede não é toda conhecida.

Para isso, pretendemos implementar os algoritmos de jornadas em Grafos Evolutivos (*Foremost Journey*, *Shortest Journey* e *Fastest Journey*) com o auxílio do simulador de redes ONE (Opportunistic Network Environment simulator) [4] e comparar sua eficiência com a dos algoritmos de roteamento comuns.

Uma vez implementados os três algoritmos de jornadas que consideram que a rede toda é conhecida, tentaremos implementá-los de forma distribuída, assumindo que cada nó tem conhecimento apenas de suas próprias conexões, um cenário mais condizente com a realidade. Para isso, tentaremos nos basear no algoritmo para calcular caminhos de custo mínimo em grafos estáticos de forma distribuída proposto por Chandy e Misra [2].

3 Atividades Realizadas

- Estudo de DTNs, Grafos Evolutivos e algoritmos de roteamento.

- Implementação dos algoritmos Foremost Journey e Shortest Journey em Java.
- Utilização do simulador ONE para verificar os resultados obtidos com os algoritmos de roteamento comuns e os algoritmos que utilizam Grafos Evolutivos.

4 Cronograma

- junho-julho/2009 - Implementação do algoritmo Fastest Journey
- agosto-outubro/2009 - Estudar a distribuição dos algoritmos
- outubro-novembro/2009 - Preparação do pôster
- novembro/2009 - Apresentação do trabalho
- julho-novembro/2009 - Elaboração da monografia

5 Estrutura da monografia

- Introdução - Apresentação do problema, motivação e objetivos da monografia.
- Conceitos e tecnologias estudadas - Definições formais dos conceitos usados e descrição dos algoritmos estudados.
- Atividades realizadas - Descrição do ambiente de simulação e das técnicas utilizadas para a implementação dos algoritmos estudados.
- Resultados e produtos obtidos - Apresentação das comparações entre os algoritmos para DTNs e os de grafos evolutivos e descrição do software desenvolvido para trabalhar com os grafos e passar estas informações para o simulador de redes.
- Conclusões - Discussão da importância dos resultados obtidos e possíveis trabalhos futuros.

Referências

- [1] J. Burgess, B. Gallagher, D. Jensen, and B.N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. In *Proc. IEEE Infocom*, pages 1–11, 2006.
- [2] K.M. Chandy and J. Misra. Distributed computation on graphs: Shortest path algorithms. 1982.
- [3] A. Ferreira. On models and algorithms for dynamic communication networks: the case for evolving graphs. In *In Proc. ALGOTEL*, 2002.
- [4] A. Keränen, J. Ott, and T. Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceeding of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.
- [5] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *Lecture Notes in Computer Science*, pages 239–254, 2004.
- [6] J. Monteiro, A. Goldman, and A. Ferreira. Using Evolving Graphs Foremost Journey to Evaluate Ad-Hoc Routing Protocols. In *In Proceedings of 25th Brazilian Symposium on Computer Networks (SBRC'07)*, Blem, Brazil, 2007.
- [7] C.T. Oliveira, M.D.D. Moreira, M.G. Rubinstein, L. Costa, and O. Duarte. Redes tolerantes a atrasos e desconexoes. In *Minicursos do Simposio Brasileiro de Redes de Computadores (SBRC 2007)*, 2007.
- [8] T. Spyropoulos, K. Psounis, and C.S. Raghavendra. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, pages 252–259. ACM New York, NY, USA, 2005.
- [9] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical report, Citeseer, 2000.
- [10] B. Xuan, A. Ferreira, and A. Jarry. Computing shortest, fastest, and foremost journeys in dynamic networks. 2002.