

Resolvendo o problema de snapshot em redes DTN utilizando algoritmos distribuídos

Maurício José Da Silva

Orientador

Ricardo Augusto Rabelo Oliveira

PPGCC, Universidade Federal de Ouro Preto

11 de julho de 2011

1 Introdução

- Algoritmos Distribuídos
- Redes DTN
- Estado global de grafos

2 Métodos

- Definir estado global
- Análise de complexidade

3 Resultados

- Experimentos

Algoritmos Distribuídos

- 1 Se comunicam por um canal que permite a comunicação ponto-a-ponto.

Algoritmos Distribuídos

- 1 Se comunicam por um canal que permite a comunicação ponto-a-ponto.
- 2 Não compartilham fisicamente qualquer tipo de memória.

Algoritmos Distribuídos

- 1 Se comunicam por um canal que permite a comunicação ponto-a-ponto.
- 2 Não compartilham fisicamente qualquer tipo de memória.
- 3 Comunicação acontece através da troca de mensagens.

Redes DTN (*Delay Tolerant Network*)

- 1 Alternativa ao TCP/IP.

Redes DTN (*Delay Tolerant Network*)

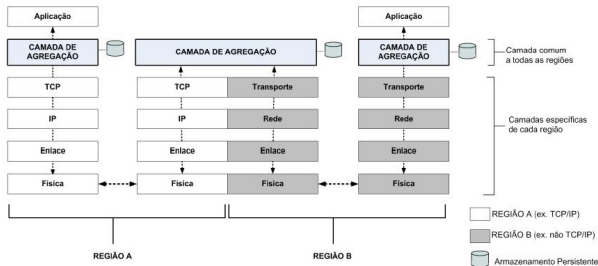
- 1 Alternativa ao TCP/IP.
- 2 Foi projetado para ser utilizado em redes Interplanetárias.

Redes DTN (*Delay Tolerant Network*)

- ① Alternativa ao TCP/IP.
- ② Foi projetado para ser utilizado em redes Interplanetárias.
- ③ Tem boa aplicação em redes de dispositivos móveis.
 - Falta de conectividade ponto a ponto.
 - Limite de armazenamento restrito.

Características de redes DTN

- 1 Nova camada denominada camada de agregação (*Bundle Layer*).
 - Armazenamento persistente.
 - Divide os pacotes em agregados (*bundles*).
 - Isola as demais camadas permitindo interoperabilidade.



Características de redes DTN

- 1 Atrasos longos: um pacote DTN não possui tempo estimado para chegar ao destino.

Características de redes DTN

- 1 Atrasos longos: um pacote DTN não possui tempo estimado para chegar ao destino.
- 2 Frequentes desconexões: as desconexões podem ocorrer pela mobilidade.

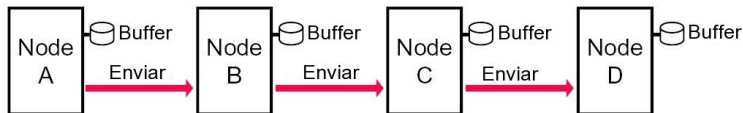
Características de redes DTN

- 1 Atrasos longos: um pacote DTN não possui tempo estimado para chegar ao destino.
- 2 Frequentes desconexões: as desconexões podem ocorrer pela mobilidade.
- 3 Capacidade de armazenamento: principal característica, armazenar e enviar.

Características de redes DTN

- 1 Atrasos longos: um pacote DTN não possui tempo estimado para chegar ao destino.
- 2 Frequentes desconexões: as desconexões podem ocorrer pela mobilidade.
- 3 Capacidade de armazenamento: principal característica, armazenar e enviar.
- 4 Transferência de custódia.

Transmissão DTN



Roteamento em redes DTN

- 1 Contato direto: envia somente se conhecer o destino.

Roteamento em redes DTN

- 1 Contato direto: envia somente se conhecer o destino.
- 2 Primeiro contato: envia para o primeiro que estabelecer conexão.

Roteamento em redes DTN

- 1 Contato direto: envia somente se conhecer o destino.
- 2 Primeiro contato: envia para o primeiro que estabelecer conexão.
- 3 Epidêmico: envia para todos os contatos.

Roteamento em redes DTN

- 1 Contato direto: envia somente se conhecer o destino.
- 2 Primeiro contato: envia para o primeiro que estabelecer conexão.
- 3 Epidêmico: envia para todos os contatos.
- 4 Probabilístico: envia para os contatos previsíveis.

Snapshot ou estado global de grafos

- 1 Técnica para registrar estados globais durante a execução de algoritmos assíncrono.

Snapshot ou estado global de grafos

- 1 Técnica para registrar estados globais durante a execução de algoritmos assíncrono.
- 2 A partir deste estado propriedades globais podem ser analisadas.

Snapshot ou estado global de grafos

- 1 Técnica para registrar estados globais durante a execução de algoritmos assíncrono.
- 2 A partir deste estado propriedades globais podem ser analisadas.
- 3 Permite a verificação de propriedades estaveis

Snapshot ou estado global de grafos

- 1 Técnica para registrar estados globais durante a execução de algoritmos assíncrono.
- 2 A partir deste estado propriedades globais podem ser analisadas.
- 3 Permite a verificação de propriedades estaveis
- 4 Estado global pode ser armazenado em G de forma distribuída.

Técnica

- 1 Nó armazena seu estado e envia uma mensagem para seus vizinhos.

Técnica

- ① Nó armazena seu estado e envia uma mensagem para seus vizinhos.
- ② Algoritmo troca mensagens entre os nós até atingir o estado global.
 - Mensagens são enviadas de forma assíncrona.
 - Cada mensagens gera um evento e altera o estado do nó.
 - Não há perda de mensagens no canal de comunicação.
 - Mensagem especial chamada de *Marker*.

Técnica

- 1 Nó armazena seu estado e envia uma mensagem para seus vizinhos.
- 2 Algoritmo troca mensagens entre os nós até atingir o estado global.
 - Mensagens são enviadas de forma assíncrona.
 - Cada mensagens gera um evento e altera o estado do nó.
 - Não há perda de mensagens no canal de comunicação.
 - Mensagem especial chamada de *Marker*.
- 3 Nó que receber a mensagem de *Marker*, registra seu estado e a reenvia.

Algoritmo em Pseudo-Código

Algoritmo *A_Record_Global_State*

▷ **Variables:**

$node_state_i = \text{nil};$

{Estado local do substrato}

$edge_state_i^j = \forall n_j \in Neig_i;$

{Estado da aresta $(n_j \rightarrow n_i)$ }

$recorded_i = \text{false};$

{Indica se o estado local já foi registrado}

$received_i^j = \text{false} \forall n_j \in I_Neig_i.$

{Indica se *marker* já foi recebido de n_j }

▷ **Input:**

$msg_i = \text{nil}.$

Action if $n_i \in N_0$:

{Estado do substrato é registrado}

$node_state_i \leftarrow \sigma_i$

{Marca que registrou o estado local}

$recorded_i \leftarrow \text{true};$

{Envia *marker*, com o estado local, em todas as arestas de saída}

Send *marker* to all $n_j \in O_Neig_i.$

Algoritmo em Pseudo-Código

Algoritmo *A_Record_Global_State*

▷ **Input:**

$msg_i = marker \mid origin_i(msg_i) = (n_j \rightarrow n_j).$

Action:

{Indica que recebeu *marker* de n_j }

$received_i^j \rightarrow \mathbf{true};$

{Faz o registro do estado local como descrito na ação (1), se ainda não fez}

if not $recorded_i$

then begin

$node_state_i \leftarrow \sigma_i$

$recorded_i \leftarrow \mathbf{true};$

Send *marker* to all $n_k \in O_Neig_i$.

end

Análise de complexidade

- 1 Algoritmos distribuídos dependem diretamente da troca de mensagens.

Análise de complexidade

- 1 Algoritmos distribuídos dependem diretamente da troca de mensagens.
- 2 Algoritmo de Snapshot
 - Tem que haver troca de mensagens entre todos os vértices.
 - Complexidade depende do número de vértices e de arestas do grafo.

$$O = n.m$$

Onde:

n = Número de vértices do grafo.

m = Número de arestas do grafo.

Análise de complexidade em grafos DTN

- 1 Deixa de ser simples em redes DTN.

Análise de complexidade em grafos DTN

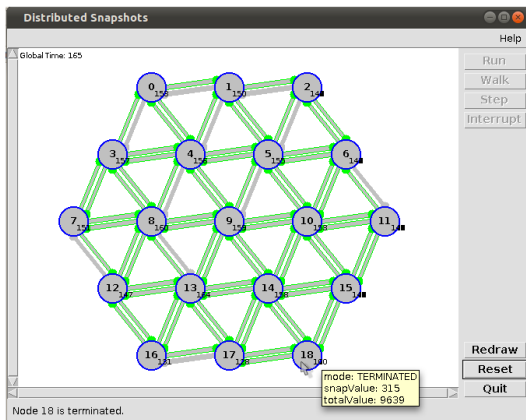
- 1 Deixa de ser simples em redes DTN.
- 2 Não podemos prever o tempo que o vértice vai ficar inalcançável.
 - Inviabiliza análise por tempo.
 - Provoca retransmissão de mensagens.

Análise de complexidade em grafos DTN

- ❶ Deixa de ser simples em redes DTN.
- ❷ Não podemos prever o tempo que o vértice vai ficar inalcançável.
 - Inviabiliza análise por tempo.
 - Provoca retransmissão de mensagens.
- ❸ Temos que considerar dois cenários.
 - Estocástico: Baseado em milhares de testes.
 - Determinístico: Baseado em um histórico.

Grafo

- 1 Testes foram realizados no simulador DAJ.
- 2 Considerando um grafo $G=(19,84)$:



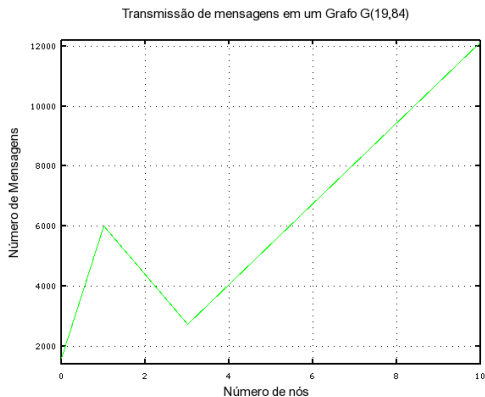
Cenário

- 1 Quatro instâncias foram resolvidas.
- 2 Os resultados são apresentados em:

Número de mensagens na rede				
Tamanho do grafo	Número de nós inativos			
	0	1	3	10
G(7,24)	168	186	363	7
G(19,84)	1596	6006	2718	12146

Grafico

1 Desvio nos resultados.



Dúvidas?