

Avaliação de Desempenho

Trabalho Prático 3 - Relatório

O algoritmo BubbleRap

O BubbleRap é um algoritmo que é usado como protocolo de roteamento. O Algoritmo baseia-se no fato de que a população carrega consigo dispositivos móveis os quais podem funcionar como nós de uma rede e fazer com que a informação se propague entre as pessoas sem a necessidade de uma infraestrutura de rede dedicada.

O conceito de comunidade é a base desse algoritmo, onde cada rede pode possuir várias comunidades e cada comunidade possui os seus nós. O critério para troca de mensagens nessa rede é a popularidade global e local de cada nó.

O algoritmo se aplica melhor a redes tolerantes a atraso, visto que o movimento das pessoas é aleatório e não há garantias que a informação siga seu caminho ao destino sem interrupções.

Análise dos algoritmos sugeridos no enunciado

1. Centralidade de Autovetor (eigenvector Centrality)

Objetivo:

Analisar a influência de um nó no funcionamento da rede.

Funcionamento:

Dado um nó e suas conexões.

Quanto mais conexões o nó possuir, maior será a sua influência.

Quanto mais conexões com nós populares o nó possuir, maior será a sua influência.

2. KClique

Objetivo:

Criar comunidades de nós dentro da rede.

Funcionamento:

Dada uma quantidade de nós e suas conexões.
Formar o maior grupo de nós em que todos estejam conectados.

3. DEGREE

Objetivo:
Saber a popularidade de um nó.

Funcionamento:
Cada nó tem um grau de conectividade.
A cada conexão com outro nó o grau de conectividade dele aumenta.

Tentativa de implementação

O que deve ser modificado no simulador:

- Criar uma configuração para o experimento.
 - AP1-avaliacao-de-desempenho-master/config_teste.txt
- Criar uma nova classe que implementa o sistema de roteamento de mensagens.
 - AP1-avaliacao-de-desempenho-master/src/routing
- Novos arquivos:
 - KCliqueCommunityDetection.java
 - Arquivo que implementa o algoritmo KClique como algoritmo para criação de comunidades. Um nó usando K-Clique mantém um registro de todos os nós que encontrou e a duração de contato cumulativo que teve com cada um. Uma vez que a duração total do contato para um desses nós excede um parâmetro configurável, o nó é adicionado ao conjunto familiar e à comunidade local do host e o conjunto familiar do nó é adicionado a uma aproximação de todos os conjuntos familiares da comunidade local do host.
 - DistributedBubbleRap.java
 - Implementação em java do algoritmo de roteamento pedido no enunciado. Em geral, cada nó mantém uma ideia de sua comunidade local, um grupo de nós que encontra com frequência. Também se aproxima de sua centralidade dentro da rede social definida por essa comunidade local e dentro da rede social global definida por todos os nós.
 - teste2.txt
 - Arquivo de configuração para que os algoritmos sejam devidamente reconhecidos pelo the one.

Tutorial para execução

1. Na pasta do projeto, com o console aberto, digite `./compile.sh` (Linux), `compile.bat`(Windows)
2. `./one.sh example_settings/teste2.txt`