

Apresentação de SDLE

1. Motivação

↳ Dizer porque é que escolhemos o algoritmo. Porque é tolerante a falhas, admite mudanças de vista.

~~↳ Dizer porque é que escolhemos o algoritmo.~~

Queríamos escolher um algoritmo de ~~com routing~~ gossip porque ~~propaga~~ a propagação de info é similar à propagação de doenças infecciosas.

2. Algoritmo - Flow Updating

(1 slide) ~~↳ Escolher~~ ^{Interessou-nos por} Flow Updating ^{convergir} ser tolerante a falhas, ~~admitir~~ na exist. mudanças de vista e convergir rapidamente em comparação com outros de mesma categoria.

↳ Explicar os algoritmos, Unicast e Broadcast

2. (cont.)

(2 slide) ↳ Funciona independentemente da topologia da rede e é robusto contra perdas de mensagens.

↳ Demarca-se de outras implementações que enviam "massa". A ideia é usar o conceito de flow de teoria de grafos. Em vez de guardar os valores numa variável ~~em~~ em cada nó, calcula-o através do valor inicial e das contribuições dos fluxos dos seus vizinhos.

↳ Foca-se em explorar a prop. de simetria do fluxo.

(3 slide) ↳ Explicar o algoritmo unicast e broadcast

3. Simulador

↳ É iniciado sendo-lhe dado os nodos, as distâncias, ~~os~~ ^{valores} para a injeção de tráfego e periodicidade de mudanças de estado.

↳ Este inicia a atividade enviando mensagens iniciais para todos os nodos, avisando que a festa vai começar.

3. (cont.)

- ↳ Procede à recolha das mensagens que cada nodo quer enviar, ~~entregando-as.~~ entregando-as.
 - ↳ De realgar que, ~~isto~~ permite a existência de mensagens de um nodo para ele próprio ^{como} ~~por~~ forma de mensagens de controlo (Timeouts).
 - ↳ Os timeouts representam fins de ronda, neste caso, o que despoleta a criação de novas mensagens. Noutros casos podem representar tempos de espera ou algo parecido.
 - ↳ Neste caso, tivemos de ~~o~~ arranjar uma forma de parar o ~~o~~ algoritmo para aferir sobre resultados.
- Assim, o simulador calcula a média das estimativas de cada ligação na totalidade dos nós e verifica se está dentro dos valores aceitáveis. ~~Acaba~~ Acaba se estiver.

4. Resultados

(1º slide) → Broadcast muito menos rondas e menos mensagens.

(1º e 2º slide) → Diferença de escalas pode ser explicada pelo facto do Unicast enviar apenas uma mensagem por ronda.

(2º slide) → Prova que mesmo com falhas, o algoritmo converge. 200 nós para este exemplo.

(1º slide) → As flutuações podem ser explicadas pelo facto de serem gerados novos grafos, o que, pela sua aleatoriedade, pode influenciar o tempo de execução do algo.

5. Conclusão

- ↳ Flow Updating é, de facto, tolerante a falhas, ao contrário de outros algoritmos de averaging.
- ↳ Em global foram atingidos os objetivos, uma vez que conseguimos implementar o algoritmo, e testá-lo com o simulador.
- ↳ Nesse sentido, o simulador foi uma ajuda preciosa, porque permitiu testar o algoritmo, mesmo em fases embrionárias da sua implementação, permitindo a correção de erros.