

## RELATÓRIO

Para atender à exigência de que, em uma rodada, todos os nodos testem os  $\log N$  clusters, utilizamos, primeiramente, de um loop que fará  $\text{ceil}(\log N)$  vezes agendamento de eventos teste para cada nodo, ou seja, um evento teste de cada nodo para cada cluster. Sendo assim, em uma rodada, todos os nodos executarão todos os testes com  $s = 1$ , depois com  $s = 2$  e assim por diante até  $s = \text{ceil}(\log N)$ . No evento repair também se faz necessário esse loop mencionado.

Os testes são feitos da seguinte forma:

- 1 - executa a função  $C(i,s)$  para o token
- 2 - para cada posição do vetor resultante da  $C(i,s)$ , de  $s^a$  posições(nodos), executa-se novamente a função  $C(i,s)$  (vamos chamá-la de  $C2(i,s)$  para melhor compreender).
- 3 - a partir da  $C2(i,s)$  do nodo 'x', o nodo token saberá se será necessário que ele teste o nodo 'x', ou não.

Como dito acima, a partir da  $C2(i,s)$  do nodo 'x', o nodo token descobre se ele testará o nodo 'x'. Esse processo ocorre da seguinte forma:

Para cada posição(nodo) desse vetor resultante da  $C2(i,s)$ , verifica se tal nodo não está falho, caso não esteja, então esse nodo é que testará o nodo 'x' (no pior caso, esse vetor será percorrido até o final, em que, com certeza, o nodo token não estará falho). Se esse nodo for o nodo token, então teste ele, do contrário, apenas siga para o próximo nodo da  $C(i,s)$ .

Esse processo se repete até que todos os nodos resultantes da  $C(i,s)$  do token sejam verificados (a ordem com que esses nodos são verificados é importante, logo o passo 2 é executado primeiramente sobre o nodo dito na primeira posição do vetor, depois sobre a segunda, e assim sucessivamente. Usa-se essa mesma abordagem para o vetor resultante da  $C2(i,s)$ , uma vez que a ordem também é importante).

Quando o nodo token testa um nodo, ele verifica se o estado daquele nodo é diferente do estado que ele tem registrado sobre o mesmo. No caso de ser diferente, ele incrementa o timestamp do nodo em um; do contrário, não altera o timestamp (ocorre uma exceção quando o timestamp sobre o nodo é -1, ou seja, ainda não foi testado nenhuma vez; nesse caso, se o nodo testado estiver falho, o timestamp é colocado como '1', do contrário '0'). No caso do nodo testado não estar falho, então atualize o timestamp do nodo token com o timestamp do nodo testado, nos casos em que o do token for menor do que o do testado.