

Aula 10 - Algoritmo Vcube

Wednesday, April 6, 2016 13:49

"Vcube: A Provably Scalable Distributed Diagnosis Algorithm"

The 5th SCALA Workshop, Supercomputing, 2014

Lembrar: HiADSD tem pior caso do número de testes $O(N^2)$

Vcube: resolve o problema

- Garante que cada nodo tem apenas 1 testador por cluster
- Outras modificações:
 - o Nodos mantêm contador EVENTOS (timestamp)
 - Inicialmente 0 -> fault-free
 - Se o timestamp é par, nodo sem-falha
 - Senão, nodo falho.
 - o Toda vez que um nodo sem-falha é testado, obtém "novidades"
 - o Testa todos os clusters em todos os intervalos: $\log N$ testes por intervalo

Algoritmo

Vcube executed at node i

```
REPEAT
  FOR  $s < -1$  TO  $\log N$  DO
    FOR all  $j \in C_{i,s}$  |  $i$  is the first fault-free node in  $C_{j,s}$  DO
      TEST  $j$ 
      IF  $j$  IS fault-free THEN
        UPDATE diagnostic information
    SLEEP until next testing interval
FOREVER
```

Teorema VCube

O número de testes executados por todos os nodos do sistema rodando o Vcube é $N \log N$ por intervalo de testes

Prova

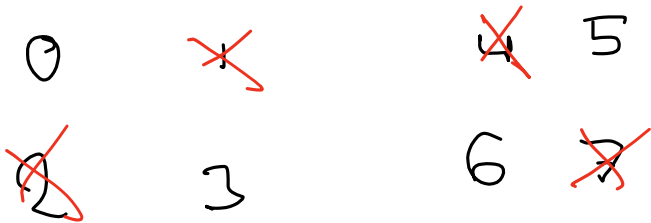
Considere os testes executados no nodo j . Para cada $C_{j,s}$, $1 \leq s \leq \log N$, o nodo j é testado pelo primeiro nodo sem-falha de $C_{j,s}$.

Cada nodo sem-falha de $C_{j,s}$ utiliza a mesma lista ordenada de nodos contendo a si mesmo | para decidir se é o testador

de j .

Portanto, cada nó j é testado por, no máximo, $\log N$ testadores por intervalo, um de cada cluster.

Exemplo Vcube



Mostre quais testes são executados neste sistema rodando o algoritmo Vcube.

Apenas os nós sem-falha são testadores:

$$\begin{aligned}
 0: C_{0,1} &= (1, 0, 1) - C_{1,1} = (0, 0, 1) \rightarrow 1 \\
 C_{0,2} &= (2, 3) - C_{2,2} = (0, 1, 3) - C_{3,2} = (1, 0, 3) \rightarrow 2, 0 \rightarrow 3 \\
 C_{0,3} &= (4, 5, 6, 7) - C_{4,3} = (0, 1, 2, 3) \rightarrow 4 \\
 C_{5,3} &= (1, 2, 3, 0) \\
 C_{6,3} &= (2, 3, 0, 1) \\
 C_{7,3} &= (3, 0, 1, 2)
 \end{aligned}$$

$$\begin{aligned}
 5: C_{5,1} &= (4) - C_{4,1} = (5) - 5 \rightarrow 4 \\
 C_{5,2} &= (6, 7) - C_{6,2} = (4, 5) - C_{7,2} = (5, 4) - 5 \rightarrow 6, 5 \rightarrow 7 \\
 C_{5,3} &= (1, 2, 3, 0) - C_{1,3} = (5, 6, 7, 4) - 5 \rightarrow 1 \\
 C_{2,3} &= (6, 7, 4, 5) \\
 C_{3,3} &= (7, 4, 5, 6) - 5 \rightarrow 3 \\
 C_{0,3} &= (4, 5, 6, 7) - 5 \rightarrow 0
 \end{aligned}$$

$$\begin{aligned}
 6: C_{6,1} &= (7) - C_{7,1} = (6) - 6 \rightarrow 7 \\
 C_{6,2} &= (4, 5) - C_{4,2} = (6, 7) - C_{5,2} = (7, 6) - 6 \rightarrow 4, 6 \rightarrow 5 \\
 C_{6,3} &= (2, 3, 0, 1) - C_{0,3} = (4, 5, 6, 7) \\
 C_{1,3} &= (5, 6, 7, 4) \\
 C_{2,3} &= (6, 7, 4, 5) - 6 \rightarrow 2 \\
 C_{3,3} &= (7, 4, 5, 6)
 \end{aligned}$$

- 3,3 \ / - - - /

Testes executados:

0 testa 1,2,3,4

3 testa 0,1,2,5,6,7

5 testa 0,1,3,4,6,7

6 testa 2,4,5,7

Exemplo 2: HiADSD

Fazer o mesmo exercício, usando o algoritmo HiADSD.

Mostre quais testes são executados e quais as info diagnóstico obtidas dos nodos testados s em-falha

0->1

0->~~2~~, 0->3

0->~~4~~, 0->5 (obtem info 6,7)

6->7

6->~~4~~, 6->5

6->~~2~~, 6->3 (obtem info 0,1)

5->~~4~~

5->~~7~~, 5->6

5->~~1~~, 5->~~2~~, 5->5 (obtem info 0)

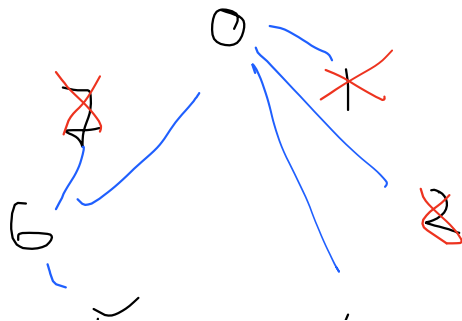
3->~~2~~

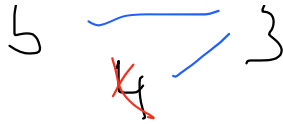
3->~~1~~, 3->0

3->~~7~~, 3->~~4~~, 3->5 (obtem info 6)

Exemplo 3: Adaptive DSD

Quais testes são executados e quais infos de diagnóstico são transferidas?





0->~~1~~, 0->~~2~~, 0->3 (obtem info sobre 4,5,6,7)

3->~~4~~, 3->5 (obtem info sobre 6,7,0,1,2)

5->6 (obtem info sobre 7,0,1,2,3,4)

6->~~7~~, 6->0 (obtem info sobre 1,2,3,4,5)