Um Algoritmo para Diagnóstico de Redes de Topologia Arbitrária

Elias Procópio Duarte Jr.
Universidade Federal do Paraná, Depto. de Informática
Caixa Postal 19081 Curitiba 81531-990 PR Brasil
e-mail: elias@inf.ufpr.br

Resumo

É crescente a demanda por sistemas de gerência de redes capazes de diagnóstico de falhas e problemas de desempenho. É importante que tais sistemas sejam, eles próprios, tolerantes a falhas. Neste trabalho, apresentamos um algoritmo para diagnóstico de falhas em redes de topologia arbitrária, aplicável a sistemas integrados de gerência. O algoritmo permite o diagnóstico de falhas nos canais de comunicação da rede, e o cálculo da conectividade sob o ponto de vista de qualquer nodo sem falhas. Trata-se de uma abordagem tolerante a falhas pois, como o algoritmo é totalmente distribuído, mesmo que ocorram falhas na rede, os nodos sem falha continúam monitorando a rede continuamente.

Abstract

There is a growing demand for network management systems that are capable of effectively diagnosing faults and performance problems. To achieve this goal, it is important that those systems themselves be fault-tolerant. In this work we present a system-level diagnosis algorithm for general topology networks, that can be applied for network fault management. The algorithm allows link fault diagnosis, after which nodes compute network connectivity. It employs the minimum number of tests, i.e. one per link per testing interval. The latency of the algorithm is proportional to the diameter of the graph corresponding to the network. This approach is fault-tolerant in the sense that no matter which portion of the network is faulty, the fault-free nodes keep on monitoring the network continuously.

1 Introdução

A constante expansão das redes de computadores dentro de empresas e organizações implica num aumento dos custos associados a eventuais falhas e problemas de performance. Ao mesmo tempo, as redes são cada vez complexas, no sentido de que se constituem de componentes heterogêneos, produzidos por uma variedade de fabricantes. Assim, é fundamental que o gerente da rede tenha a sua disposição um conjunto de ferramentas que lhe permitam evitar problemas, e resolvê-los com rapidez se por ventura ocorrerem.

A maioria dos sistemas de gerência de rede atuais se baseia no protocolo SNMP ("Simple Network Management Protocol") [3]. Nestes sistemas, uma máquina é responsável por supervisionar toda a rede. Esta máquina é chamada *NMS* ("Network Management Station"), e em geral apresenta uma interface gráfica através da qual o gerente humano tem acesso ao sistema. Para monitorar a rede, o NMS se comunica com uma série de *agentes*, cada um deles responsável por monitorar um componente da rede, que pode ser uma máquina, um canal de comunicação, um hub, um protocolo, entre outros. A comunicação entre NMS e agentes se dá através de um protocolo de gerência de redes, como o SNMP. O NMS pode realizar um polling periódico dos agentes, ou então receber "traps", quer dizer, interrupções que informam situações de emergência.

Os sistemas centralizados apresentam dois problemas: em primeiro lugar, se o NMS falhar, então a rede deixará de ser monitorada. Além disso, outro problema é a concentração de mensagens de gerência em um único nodo, o que pode acarretar efeitos negativos na sua performance. Neste trabalho descrevemos um algoritmo totalmente distribuído e tolerante a falhas para monitorar uma rede de topologia arbitrária.

Esta estratégia é baseada na teoria de diagnóstico de falhas a nível de sistema ("system-level diagnosis"). Esta teoria já vem se desenvolvendo há mais de 30 anos. Os algoritmos práticos para diagnóstico a nível de sistema podem ser divididos em duas categorias: os que assumem que entre todo par de nodos na rede existe um único canal de comunicação, quer dizer, que o grafo do sistema é completo, e os que permitem que a rede tenha topologia arbitrária. O Hi-ADSD é um algoritmo recente para diagnóstico de redes totalmente conexas [1].

O algoritmo NBND ("Non-Broadcast Network Diagnosis") descrito neste trabalho foi introduzido inicialmente em [2]. Ele permite o diagnóstico de falhas nos canais de comunicação da rede e o cálculo da conectividade sob o ponto de vista de qualquer nodo sem falhas. Trata-se de uma abordagem tolerante a falhas pois, como o algoritmo é totalmente distribuído, mesmo que ocorram falhas na rede, os nodos sem falha continúam monitorando a rede continuamente. O algoritmo emprega o menor número de testes possível, e apresenta latência igual ao diâmetro do grafo que representa a rede.

O resto do trabalho está estruturado da seguinte forma. Na seção 2, é feita uma revisão de algoritmos para diagnóstico em redes de topologia arbitrária. Na seção 3 o algoritmo NBND é descrito, junto com um exemplo de execução. A seção 4 conclui o trabalho.

2 Algoritmos para Diagnóstico a Nível de Sistema

Considere um sistema que consiste de N unidades, ou nodos, que podem estar *falhos* ou *sem-falha*. O objetivo dos algoritmos de diagnóstico a nível de sistema é que todos os nodos sem-falha determinem o estado de todos os nodos do sistema. Podem ocorrer dois tipos de *evento* num sistema: um nodo sem-falha se tornar falho, e um nodo falho se tornar sem-falha. Cada nodo executa testes em outros nodos, e os nodos sem-falha são capazes de determinar corretamente o estado dos nodos testados por eles.

Se pensarmos em cada teste como uma aresta direcionada do testador para o nodo testado, então o conjunto de todos os nodos e todos os testes é o chamado "grafo de

testes".

Os algoritmos para diagnóstico são executados em etapas, ou *rounds*. Para entender o conceito de etapa, é necessário entender antes o conceito de intervalo de testes. Um nodo executa testes a cada intervalo de testes, que pode ser, por exemplo, 10 segundos. Quando todos os nodos sem-falha tiverem executado seus testes, uma etapa se completou. O número de etapas necessárias para que todos os nodos sem-falha do sistema façam o diagnóstico de um determinado evento é uma medida importante da eficiência de um algoritmo de diagnóstico, e é chamada *latência* do algoritmo.

2.1 Algoritmos para Redes de Topologia Arbitrária

Existem duas categorias principais de algoritmos de diagnóstico. Uma categoria realiza diagnóstico em redes nas quais existe um canal de comunicação direto entre quaisquer dois nodos da rede. A outra categoria admite que a topologia seja arbitrária, quer dizer, entre dois nodos do sistema pode haver ou não um canal de comunicação direto. No caso de não haver, para que os dois nodos se comuniquem, eles devem usar nodos intermediários.

Em [4], Bagchi e Hakimi introduziram um algoritmo para diagnóstico em redes de topologia arbitrária que utiliza o menor número possível de mensagens. Os nodos sem-falha formam um grafo de testes, através do qual as mensagens de diagnóstico são transmitidas. Entretanto, este algoritmo é executado off-line: ele não permite um monitoramento dinâmico e contínuo da rede.

Em [5] Bianchini e outros introduziram e simularam o algoritmo "Adapt". Este algoritmo é executado on-line: quando ocorre um evento, os nodos sem-falha se reorganizam de forma a manter o grafo de testes conexo, se possível. Este grafo de testes é o menor grafo fortemente conexo cujos nodos são os nodos sem falha, e cujas arestas são os canais de comunicação entre tais nodos. Para construir o grafo de testes, o algoritmo emprega um procedimento distribuído que requer grandes quantidades de mensagens enormes.

Recentemente, Rangarajan e outros introduziram um novo algoritmo para diagnóstico em redes de topologia arbitrária [6]. Este algoritmo, que chamaremos de RDZ (das iniciais dos autores), é também executado on-line, como o algoritmo Adapt. Além disso, o grafo de testes garante o menor número de testes por nodo, quer dizer, cada nodo é testado por apenas um único testador. O algoritmo apresenta a melhor latência, decorrente de uma estratégia paralela de disseminação das mensagens de diagnóstico. Entretanto, certas configurações de falhas constituem eventos que são diagnosticados apenas *eventualmente*, quer dizer, não há uma garantia de quando estes eventos são diagnosticados. Este problema impede o uso do algoritmo RDZ para gerência de falhas de redes.

3 O Algoritmo NBND

Nesta seção descrevemos o algoritmo NBND ("Non-Broadcast Network Diagnosis"). Este algoritmo foi desenvolvido para ser aplicado a sistemas de gerência de falhas em redes. Trata-se de uma estratégia que permite o diagnóstico de time-outs de canais de comunicação, e calcula conectividade dos nodos da rede, usando o menor número de testes

possível: um por canal. Além disso, o algoritmo apresenta a melhor latência possível, proporcional ao diâmetro da rede.

O algoritmo NBND se baseia no fato de que é possível determinar se um determinado canal está dando "time-out" ou se está sem-falha. Entretanto, é impossível distinguir se um nodo está falho ou se todos os canais de comunicação para tal nodo é que estão falhos. Como mostra a figura 1, estas configurações de falhas são ambíguas. Assim, ao invés de calcular quais nodos estão falhos e quais nodos estão sem falhas, o algoritmo calcula quais nodos estão atingíveis e quais nodos estão estão inatingíveis. Resumindo, os estados de um canal de comunicação são *sem-falha* e *timed-out* e os estados de um nodo são *sem-falha* ou *inatingível.*

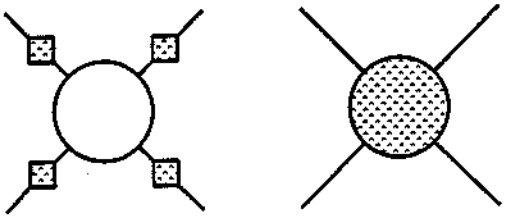

Figura 1: Configurações de falhas ambíguas.

O algoritmo utiliza o menor número possível de testes: cada canal de comunicação é testado por apenas um dos nodos por ele interligados. A estratégia proposta em [2] é a escolha do nodo com maior identificador como testador de um canal específico. No caso deste nodo, ou do próprio canal de comunicação, se tornar falho, o nodo de menor identificador passa então a executar testes. É importante observar que esta distribuição de testes pode causar situações em que alguns nodos executem diversos testes, enquanto outros executam poucos. Uma estratégia mais justa é os os nodos se alternarem como testadores e testados a cada intervalo.

Cada nodo mantém uma tabela com o estado de todos os nodos da rede. Esta tabela é inicializada em zero, e incrementada de uma unidade cada vez que o nodo sofrer um evento. Assim, valores pares indicam um nodo sem falhas, e valores ímpares indicam nodos falhos.

Quando um novo evento é descoberto, o nodo testador dissemina a informacao para todos seus vizinhos em paralelo, e estes, por sua vez, repetem o processo. Para reduzir o número total de mensagens, estas contêm um campo no qual os identificadores de nodos que já sabem do novo evento são armazenados. As mensagens recebidas podem ser opcionalmente processadas apenas uma vez a cada intervalo de testes. Neste caso, elas devem ser armazenadas em uma tabela, na medida em que são recebidas.

Após receber informação sobre um novo evento, cada nodo executa um algoritmo para determinar a conectividade da rede, quer dizer, se esta está particionada ou não, e em que pontos. Em resumo, o algoritmo é como apresentamos a seguir:

```
BEGIN
 /* at nodo i */
 DO FOREVER
  FOR each link i-j, that connects node i to node j
    IF my_turn(j) /* computes based on last message received from j */
    THEN test link i-j;
         IF link i-j is fault-free
         THEN i and j exchange messages;
         IF there is a new event
         THEN update link status; disseminate messages;
  compute node reachability;
  SLEEP(testing interval)
END;
```

3.1 Exemplo de Diagnóstico

Nesta seção apresentamos um exemplo da execução do algoritmo NBND para a rede da figura 2. Em cada aresta do grafo, os arcos estão direcionados do nodo testador para o nodo testado. Inicialmente todos os nodos estão livres de falhas.

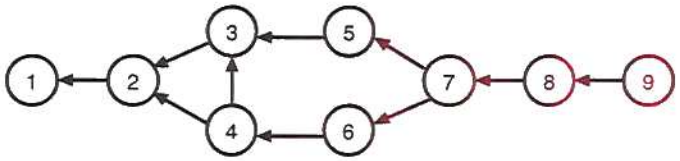

Figura 2: Grafo de testes da rede do exemplo.

Considere o seguinte evento nesta rede: o canal de comunicação entre o nodo 3 e o nodo 5 fica falho no tempo t=100. No próximo intervalo de testes, o nodo 5 detecta o evento, é o instante de tempo t=120. Ele dissemina mensagens para seu vizinho sem-falha, o nodo 7. Este processa a informacao em t=150, e dissemina para seus vizinhos, os nodos 6 e 8. Neste intervalo, como o nodo 3 não recebeu informações sobre o canal que o liga ao nodo 5, ele o testa, e descobre o evento. A informacao é disseminada para os nodos 2 e 4. O nodo 4 recebe mensagem tambem do nodo 6, referente ao mesmo evento. Neste momento, o nodo 9 recebe mensagem de diagnóstico do nodo 8. Os nodos 4 e 9 nao disseminam mensagens, pois sabem que tanto o nodo 8, como 2 e 6 já sabem do evento. Em t=180, o nodo 1 recebe a mensagem do nodo 2, completando o diagnóstico. São necessários, ao todo, 3 intervalos de testes para o diagnóstico completo. São disseminadas 8 mensagens sobre o evento. Apenas o nodo 6 recebe duas mensagens sobre o mesmo evento.

4 Conclusões

Neste artigo apresentamos um algoritmo para diagnóstico de redes de topologia arbitrária. O propósito do algoritmo é ser aplicado para sistemas de gerência de redes de computadores. Além de apresentar a melhor latência possível, proporcional ao diâmetro da rede, o algoritmo emprega o menor número de testes, um por canal de comunicação por intervalo de testes. Ao descobrir um novo evento, cada novo dissemina mensagens de diagnóstico em paralelo para seus vizinhos. São previstos mecanismos para evitar certas mensagens redundantes. Espera-se para breve resultados de simulações do algoritmo sobre redes de diversas tecnologias. Além disso, o algoritmo está sendo implementado integrado a uma sistema de gerência de redes baseado em SNMP.

Referências

[1] E.P. Duarte Jr., and T. Nanya, "A Hierarchical Adaptive Distributed System-Level Diagnosis Algorithm," *IEEE Transactions on Computers*, pp.34-45, Vol.47, No.1, Jan 1998.

[2] E.P. Duarte Jr., G. Mansfield, T. Nanya, and S. Noguchi, "Non-Broadcast Network Fault Monitoring Based on System-Level Diagnosis," *Proc. IEEE/IFIP IM'97*, pp.597-609, San Diego, May 1997.

[3] M.T. Rose, *The Simple Book - An Introduction to Internet Management*, 2^{nd} ed., Prentice-Hall, Englewood Cliffs, NJ, 1994.

[4] A. Bagchi, and S.L. Hakimi, "An Optimal Algorithm for Distributed System-Level Diagnosis," *Proc.* 21^{st} *Fault Tolerant Computing Symp.*, June, 1991.

[5] M. Stahl, R. Buskens, and R. Bianchini, "Simulation of the Adapt On-Line Diagnosis Algorithm for General Topology Networks," *Proc. IEEE* 11^{th} *Symp. Reliable Distributed Systems*, October 1992.

[6] S.Rangarajan, A.T. Dahbura, and E.A. Ziegler, "A Distributed System-Level Diagnosis Algorithm for Arbitrary Network Topologies," *IEEE Transactions on Computers*, Vol.44, pp. 312-333, 1995.