Uma Arquitetura de Balanceamento de Carga em Sistemas Distribuídos Utilizando Redes Neurais Artificiais

Andreia Carniello
ancarnie@lac.inpe.br
Instituto Nacional de Pesquisas Espaciais

Maurício G. V. Ferreira
mauricio@ccs.inpe.br
Instituto Nacional de Pesquisas Espaciais

José Demisio Simões da Silva
demisio@lac.inpe.br
Instituto Nacional de Pesquisas Espaciais

Resumo

Uma questão importante em sistemas distribuídos é o gerenciamento de carga dos nós do sistema. O balanceamento de carga possibilita um melhor aproveitamento das capacidades computacionais da rede e um ganho de desempenho por meio da alocação de nós mais adequados a execução de determinadas tarefas. Este trabalho propõe uma arquitetura de balanceamento de carga em sistemas distribuídos – BalRN, que utiliza redes neurais artificiais e um conjunto de políticas como suporte ao processo de migração de objetos.

***Palavras-chave**: sistemas distribuídos, balanceamento de carga, redes neurais artificiais*

1. Introdução

As principais vantagens dos sistemas distribuídos são o alto desempenho, a disponibilidade de recursos e a extensibilidade a um baixo custo. Em um sistema distribuído típico as tarefas chegam aos nós de forma aleatória. Isso pode gerar uma situação de balanceamento não-uniforme entre os nós do sistema. Esse desbalanceamento resume-se na existência de nós com altas cargas e outros nós levemente carregados ou muitas vezes ociosos. Essa situação é prejudicial ao sistema em termos do tempo de resposta das tarefas e da utilização de recursos [1].

O serviço de balanceamento precisa de informações que o auxilie nas decisões de como distribuir a carga do sistema de forma uniforme. Para auxiliar as decisões do serviço de balanceamento, técnicas de inteligência artificial podem ser empregadas. Uma técnica favorável neste processo são as redes neurais artificiais, as quais empregam uma interligação maciça de células computacionais chamadas de "neurônios" utilizadas na resolução de problemas de classificação e de otimização.

Neste trabalho as redes neurais artificiais são utilizadas como suporte ao serviço de balanceamento, pois as redes provêem informações ao serviço de balanceamento para que este tome decisões de alocação de recursos.

Este trabalho pretende aplicar o serviço de balanceamento de carga proposto ao protótipo do sistema de software utilizado para controle de satélites. A idéia é incorporar o serviço de balanceamento proposto ao protótipo do sistema de software de controle de satélites para fazer realocação dos objetos da aplicação em tempo real, conforme o aumento dos pedidos de serviço.

2. Arquitetura BalRN

Neste trabalho é proposta uma arquitetura de balanceamento de carga: a arquitetura BalRN, conforme mostra a Figura 1. Esta arquitetura é formada por dois componentes: Serviço de Redes Neurais Artificiais e Serviço de Balanceamento de Carga. O Serviço de Balanceamento de Carga toma suas decisões com base nas informações providas pelo Serviço de Redes Neurais Artificiais, as quais coletam índices de carga (uso de CPU e de memória) para conhecerem o estado real dos nós do sistema.

O Serviço de Balanceamento de Carga é quem define os nós transmissores e receptores de carga, quem atualiza o estado de carga de todo o sistema, seleciona os objetos a serem migrados e os nós receptores destes objetos. Isto é feito por meio das Políticas de Migração ([3] e [4]): Política de Transferência, Política de Informação, Política de Seleção e de Localização, respectivamente. O Executor recebe informações provenientes das políticas e toma decisões sobre quando realizar o balanceamento e é o responsável por realizar a migração propriamente dita. O Serviço de Balanceamento atua na Aplicação Distribuída de forma a remanejar os objetos da aplicação com o objetivo de obter uma distribuição de carga uniforme em todo o sistema.

O Serviço de Redes Neurais Artificiais é composto por três redes neurais: Classificação de Carga, Hopfield e Aprendizagem. A rede de Classificação de Carga é responsável por determinar o estado de carga dos nós. Esta rede coleta dois índices de carga: uso de CPU e de memória, e classifica os nós em cinco possíveis estados: muito ocioso, pouco ocioso, normal, pouco carregado e muito carregado. Esta informação de classificação é utilizada pela Rede de Hopfield [2], uma rede neural que trabalha com base no conceito de energia com o objetivo de minimizá-la, gerando para o sistema uma configuração com equilíbrio de carga entre os nós. Esta configuração ideal de carga gerada pela rede de Hopfield é utilizada pelo Executor de balanceamento de carga em suas tomadas de decisão. Por fim, a rede de Aprendizagem é uma rede capaz de aprender a partir do funcionamento da rede de Hopfield e, por isso, pode substituí-la. Esta substituição diminuiria o custo

do Serviço de Redes Neurais, dado o alto custo associado às redes de Hopfield.

3. Funcionamento da Arquitetura BalRN

No momento que o serviço de balanceamento de carga é ativado, todos os nós do sistema coletam seus índices de carga: uso de CPU e de memória. Estas informações são utilizadas pela rede neural de Classificação de Carga (presente em cada nó do sistema) para determinar o nível de carga dos nós, sendo cinco os níveis considerados neste trabalho: muito ocioso, pouco ocioso, normal, pouco carregado e muito carregado.

Em seguida, coleta-se o estado de carga de todos os nós do sistema para que o nó que ativou o serviço de balanceamento tenha informações reais de carga dos demais nós. O Executor verifica, então, se há necessidade de realizar balanceamento de carga. Se existirem nós "muito carregado", então, a carga do sistema será balanceada. Caso contrário, não haverá o balanceamento. Neste caso, o sistema aguarda por um Δt para ativar novamente o serviço de balanceamento em um próximo nó, segundo uma fila circular dos nós do sistema.

Quando o Executor identifica a necessidade de realizar o balanceamento, a rede de Hopfield é acionada para minimizar a energia do sistema e, com isso, produzir uma configuração de carga ideal para os nós do sistema. O processo de migração inicia, então, com o objetivo de atingir este estado de equilíbrio de carga proposto pela rede de Hopfield. Seleciona-se os objetos a serem migrados (a partir dos nós "muito carregado") e os nós receptores destes objetos (a partir dos nós "muito ocioso") e, então, procede-se com a migração propriamente dita.

Ocorrida a migração, é preciso analisar o estado de carga dos nós. A rede de Classificação de Carga é acionada novamente e as informações de carga são atualizadas no sistema. O Executor, então, verifica se a nova configuração de carga gerada após a migração está próxima da configuração gerada pela rede de Hopfield. Se sim, então, o sistema atingiu equilíbrio de carga e encontra-se balanceado. Caso contrário, o sistema não está balanceado ainda e uma nova configuração de carga é gerada pela rede de Hopfield. Procede-se com uma nova migração de objetos até que a configuração de carga real do sistema se aproxime da gerada pela rede de Hopfield.

Todos os nós do sistema têm autonomia para realizar o balanceamento de carga, assim, a ativação do serviço de balanceamento pode ser feita por qualquer nó. Assume-se a existência de um *token* que percorre os nós e determina qual o nó que ativará o serviço de balanceamento. O percurso deste *token* obedece a uma fila circular dos nós do sistema. Assim, o serviço de balanceamento é ativado em cada momento por um nó distinto e esta ativação ocorre conforme um Δt pré-estabelecido.

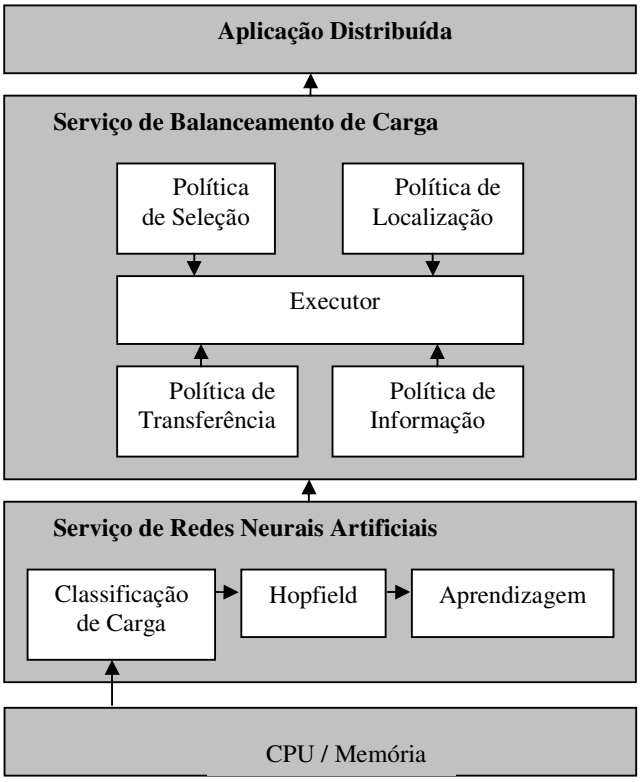

Figura 1 - Arquitetura de Balanceamento de Carga BalRN

A política de informação adotada é distribuída e periódica, ou seja, os nós coletam informações sobre todo o sistema periodicamente. Esta coleta de informações de carga sobre os outros nós do sistema é feita a cada ativação do serviço de balanceamento (que obedece a um Δt pré-estabelecido).

4. Políticas de Migração

Nesta seção apresentam-se as políticas de transferência, de seleção e de localização propostas.

4.1. Política de Transferência

A política de transferência identifica se um nó está em um estado adequado para participar de uma migração, como transmissor ou como receptor.

A política de transferência proposta utiliza dois índices de carga para definir o nível de carga dos nós do sistema. Esses índices expressam a taxa de utilização de CPU e o uso de memória. Para definir o nível de carga de um nó *h*, definiu-se a seguinte heurística: analisa-se os dois índices de carga considerados e classifica-se o nó *h* em um dos cinco níveis de carga: nó muito ocioso, pouco ocioso, normal, pouco carregado e muito carregado.

A política de transferência é implementada por meio de uma rede neural artificial de arquitetura MLP (*Multiple Layer*

Perceptron). Esta rede realiza o treinamento de forma supervisionada pelo algoritmo de retropropagação de erro [2].

4.2. Política de Seleção

Objetos em nós "muito carregado" devem ser migrados para nós "muito ocioso". A política de seleção escolhe o objeto a ser migrado. Nossa proposta é analisar quatro fatores para a escolha do objeto a ser migrado. Esses fatores são descritos a seguir.

Considera-se que um objeto *x* localiza-se fisicamente em um nó *h* "muito carregado". Considera-se também que:

$L_h(x)$ é o conjunto de objetos localizados fisicamente no nó *h* que se relacionam com o objeto *x*;

$R_h(x)$ é o conjunto de objetos remotos ao nó *h* que se relacionam com o objeto *x*;

$T(x)$ é o custo de transferência do objeto *x* (tamanho do objeto, em bytes);

$RL(x)$ é a quantidade de relacionamentos entre o objeto *x* e objetos *i*, sendo que $i \in L_h(x)$;

$RR(x)$ é a quantidade de relacionamentos entre o objeto *x* e objetos *j*, sendo que $j \in R_h(x)$;

$C(x)$ é a quantidade de conexões ativas do objeto *x* com objetos *k*, sendo que $k \in L_h(x)$ ou $k \in R_h(x)$.

Para os objetos localizados em um nó *h* (classificado pela política de transferência como "muito carregado"), a política de seleção analisa qual objeto deve ser migrado/replicado. A escolha do objeto é feita conforme a seguinte heurística: migra-se o objeto que apresentar menor custo de transferência $-T(x)$, menor quantidade de relacionamentos com objetos do nó $h - RL(x)$, maior quantidade de relacionamentos com objetos remotos ao nó $h - RR(x)$ e que não apresente conexão ativa com outro objeto $-C(x)$. Caso $C(x)>0$, o objeto *x* não deve ser migrado e sim replicado para o nó receptor de carga.

4.3. Política de Localização

A política de localização é responsável por selecionar os nós receptores de carga. Nossa proposta baseia-se na análise de dois fatores para a escolha do nó receptor de carga, descritos a seguir.

Considera-se que *x* é o objeto a ser migrado, *h* é um nó "muito ocioso" do sistema e *m* é o nó onde reside o objeto *x* a ser migrado. Considera-se também que:

$RL(x)$ é a quantidade de relacionamentos entre o objeto *x* migrado e objetos *i*, sendo que $i \in L_h(x)$;

$D(m,h)$ é a distância do nó *m* ao nó *h* medida pelo tempo de resposta ao comando *ping*.

Para a migração/replicação de um objeto *x*, a política de localização analisa para qual nó "muito ocioso" do sistema o objeto *x* deve ser migrado/replicado. A heurística adotada é a seguinte: migra-se o objeto *x* para o nó que tenha a maior quantidade de relacionamentos com o objeto *x* – $RL(x)$ e a menor distância com relação ao nó onde reside o objeto *x* $D(m,h)$.

5. Conclusões

Em sistemas distribuídos, dada a alta disponibilidade que se deseja alcançar no sistema, não é interessante que o sistema de balanceamento de carga seja centralizado, pois isso implica em um ponto único de falha no sistema. Uma alternativa mais interessante, do ponto de vista científico, é construir um sistema de balanceamento de carga que seja distribuído, ou seja, um sistema no qual as decisões de escalonamento podem ser tomadas por qualquer nó do sistema, sem haver um controle centralizado.

A arquitetura de balanceamento de carga proposta – BalRN incorpora este conceito, uma vez que é concedida a cada nó do sistema autonomia para realizar o balanceamento de carga, o que elimina o gargalo de se ter um escalonador único e aumenta a disponibilidade do sistema no caso de falha, pois a falha de um nó não compromete o serviço de balanceamento como um todo.

Assim, a arquitetura BalRN realiza o balanceamento de carga de forma distribuída e dinâmica, redistribuindo a carga entre os nós do sistema em tempo de execução. Esta redistribuição é feita com base no conjunto de políticas definido na arquitetura, o qual estabelece como a migração dos objetos deve ser conduzida. Estas políticas de migração utilizam informações para tomar decisões de balanceamento, as quais são providas pelas redes neurais artificiais por serem indicadas na resolução de problemas de classificação e de otimização.

Assim, este trabalho propõe uma arquitetura de balanceamento de carga em sistemas distribuídos utilizando redes neurais artificiais e objetiva aplicar as idéias propostas em um sistema que executa uma aplicação de objetos distribuídos, como é o caso do software Simulador de Controle de Satélites do INPE.

6. Referências

[1] EL-ABD, A., EL-BENDARY, M., **Neural-Based Selection and Location Policies for Dynamic Load Balancing in Distributed Computing Systems**, Proceedings of the IASTED International Conference on Modeling and Simulation, May, 1998.

[2] HAYKIN, S., **Redes Neurais - Princípios e Prática**, 2 ed., Bookman, 2001.

[3] SHIVARATRI, N.G., KREUGER, P., SINGHAL, M. **Load Distribution for Locally Distributed Systems**, *Computer*, v. 25, pp. 33-44, Dezembro, 1992.

[4] SONG, J.; CHOO, H.K.; LEE, K.M. **Application-Level Load Migration and its Implementation on top of PVM**. Concurrency: Practice and Experience, v. 9, pp. 1-19, 1997.