

Rebalanceamento de Carga em uma Aplicação BSP Usando as Métricas: Processamento, Comunicação e Memória

Rodrigo da Rosa Righi¹, Alexandre Carissimi¹, Philippe Olivier Alexandre Navaux¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brasil

{rodrigo.righi,asc,navaux}@inf.ufrgs.br

1. Introdução

A computação distribuída envolve cada vez mais recursos dinâmicos e aplicações irregulares. Nesse sentido, a carga acaba sendo uma medida variável e um mapeamento inicial de processos para recursos pode não permanecer eficiente no decorrer do tempo. Isso porque tanto o estado dos recursos quanto o da rede podem variar na execução da aplicação, bem como a quantidade de processamento e a interação pela rede entre os processos dela. Assim, o reescalonamento (migração) de processos para novos processadores torna-se pertinente para melhorar o uso dos recursos e o tempo de uma aplicação. Além da carga, nesse tema deve-se observar a memória do processo e as operações de migração para decidir sobre a viabilidade da transferência de um processo. Nesse contexto, esse artigo mostra algumas idéias de um modelo de rebalanceamento dinâmico de carga que **controla o reescalonamento de processos** de uma aplicação BSP (*Bulk-Synchronous Parallelism*) [Righi et al. 2007]. Segundo Camargo et al., o BSP representa um modelo de programação atrativo para a computação dinâmica e heterogênea [Camargo et al. 2005].

2. Modelo de Rebalanceamento de Carga

O rebalanceamento de carga é feito através de um reescalonamento dinâmico de processos, onde as informações são capturadas em tempo de execução da aplicação. A idéia do modelo proposto é oferecer um controle da migração de processos BSP, trabalhando nas partes de computação local e de comunicação de cada super-etapa da aplicação. A Figura 1 (a) ilustra uma super-etapa i sem balanceamento de carga. O desempenho fica comprometido uma vez que a barreira espera pelo processo mais devagar. Já a Figura 1 (b) mostra o resultado esperado com o rebalanceamento de carga no final da super-etapa i , que pode ser visto na super-etapa seguinte, $i + 1$. A arquitetura alvo é formada por agregados e redes locais potencialmente heterogêneos usados de forma não dedicada. Em adição, ela possui uma noção de hierarquia em dois níveis, com conjuntos (computadores da mesma rede) e gerentes de conjunto. O modelo oferece algoritmos que decidem as seguintes questões sobre migração: (i) Quando realizar o remapeamento de processos; (ii) Quais processos são candidatos para a migração; (iii) Qual o destino de cada processo candidato para migrar. A resposta da questão (i) é dada no final de uma super-etapa, após a barreira de sincronização. Os resultados do reescalonamento podem ser vistos nas super-etapas seguintes. Esse momento foi escolhido porque nele tem-se dados de processamento e de comunicação dos processos, possibilitando uma tomada de decisão mais correta. O tratamento dessa questão usa um índice α que determina o intervalo entre super-etapas para o próxima ativação do rebalanceamento. O valor de α aumenta caso o sistema está estável (sem grandes disparidades de tempo entre os processos). Em contra-partida, α diminui caso o sistema apresentar instabilidade.

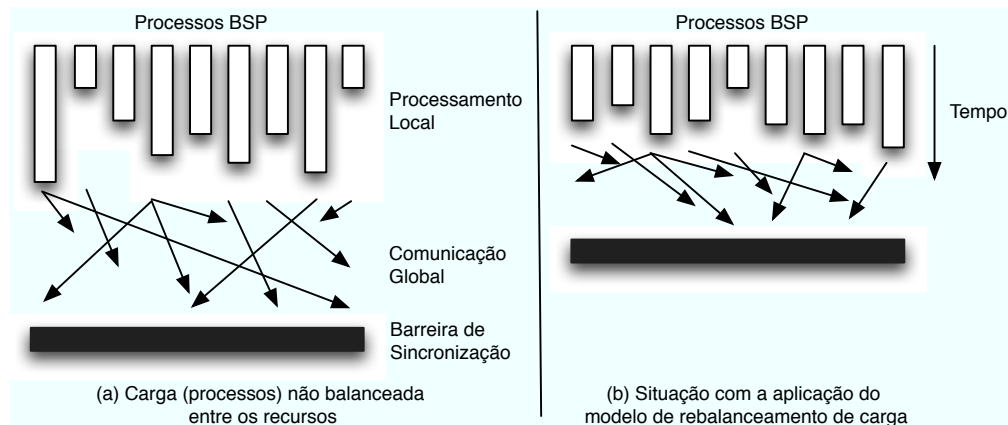


Figura 1. (a) Carga não balanceada entre os recursos na super-etapa i ; (b) Resultado do rebalanceamento de carga pode ser visto numa super-etapa seguinte

A segunda questão é resolvida através do cálculo do Potencial de Migração. Cada processo i calcula n funções $PM(i, j)$, onde n é o número de conjuntos do sistema e j simboliza um determinado conjunto. $PM(i, j)$ é encontrado através das métricas Processamento, Comunicação e Memória e a relação delas é baseada na idéia de Força da Física. Processamento e Comunicação são métricas que atuam como forças a favor da migração, enquanto a Memória representa uma idéia de custo e atua em sentido contrário. Um processo i com $PM(i, j)$ alto significa que ele possui elevado tempo de processamento e de comunicação com processos do conjunto j e possui um baixo custo de migração. Existem três heurísticas para a escolha dos candidatos a migração, todas elas baseadas no maior valor de PM de cada processo. Elas são: (i) escolha de n processos; (ii) escolha de uma percentagem de processos; (iii) seleção do processo com maior PM. O $PM(i, j)$ de um processo i candidato está associado a um conjunto j e o gerente dele é quem escolhe o melhor processador sobre sua jurisdição para receber esse processo. Logo após, verifica-se a viabilidade da migração para o processador escolhido levando em conta a carga externa nos processadores origem e destino, bem como os eventuais processos BSP que eles já possuem, e o custo da migração do processo em pauta. Por fim, para cada processo BSP candidato é escolhido um novo recurso ou a migração dele é abortada.

3. Conclusão e Trabalhos Futuros

O modelo proposto oferece rebalanceamento de carga em uma aplicação BSP considerando as três métricas estudadas. Ele também leva em consideração a questão da adaptação quanto a ativação do rebalanceamento e verifica a viabilidade de cada migração. A carga de trabalhos futuros tem-se a avaliação do modelo com o simulador Simgrid.

Referências

- Camargo, D., Kon, F., and Goldman, A. (2005). Portable checkpointing and communication for bsp applications on dynamic heterogeneous grid environments. In *17th Symposium on Computer Architecture on High Performance Computing*, p. 226–234.
- Righi, R., Carissimi, A., and Navaux, P. O. A. (2007). On the dynamic load-rebalancing of bsp application using process miration. In *V Workshop em Processamento Paralelo e Distribuído (WSPPD 2007)*, pages 31–36.