

Considerações Sobre o Ensino de Alto Desempenho

**Simone Ferreira, Mônica Py, Elias Carvalho,
Prof. Tiarajú A Diverio**

Instituto de Informática e PPGC da UFRGS
Caixa Postal 15064 91501-970 Porto Alegre – RS - Brasil
E-mail: {simonepf, mpy, diverio}@inf.ufrgs.br
<http://www.inf.ufrgs.br/~hgmc>

Introdução

Em vista da importância crescente da Teoria da Computação, cujo desenvolvimento é motivado pelas necessidades da Ciência da Computação e pelos avanços tecnológicos, busca-se desenvolver o raciocínio necessário para a formalização do Processamento de Alto Desempenho.

Atualmente, usuários de computadores e da Internet têm disponíveis cada vez mais recursos, máquinas mais poderosas e com menor custo, nas quais se pode resolver problemas maiores e cada vez mais rápido. Uma dificuldade reside na falta de experiência dos usuários para resolverem seus problemas nesses ambientes. Em sua formação básica, nada lhes foi ensinado sobre processamento de alto desempenho. Isso descreve um quadro no qual: recursos estão disponíveis; problemas exigem que as soluções sejam obtidas mais rapidamente; os usuários não estão preparados para esse uso; as universidades não estão preparando seus alunos para essa nova realidade.

Foster [1] define que um computador paralelo é um conjunto de processadores que são capazes de trabalhar cooperativamente para resolver um problema computacional. Essa definição é ampla o suficiente para incluir supercomputadores que têm centenas ou milhares de processadores, redes de estações de trabalho e estações de trabalho multiprocessadoras. Um estudo de tendências em aplicações, de arquiteturas de computadores e de rede de computadores mostra que a computação paralela está se tornando uma questão essencial nas resoluções de problemas.

Programação de Alto Desempenho (PAD)

Existe um crescente avanço da oferta de recursos de alto desempenho no mercado, disponibilizando cada vez mais recursos de alto desempenho, como: supercomputadores, redes de computadores, aglomerados de computadores e, até mesmo, computadores pessoais com mais de um processador, cujo desempenho ultrapassa, em muito, o desempenho de vários supercomputadores do passado.

Os Clusters são aglomerados de estações (por exemplo, PCs) interligados por redes de alto desempenho, que transformam várias máquinas em um único ambiente de programação concorrente; existem também ambientes de programação distribuída através de redes de computadores; e finalmente centros de supercomputação. Há, entretanto, falta de profissionais qualificados na área, o que tem contribuído para que a demanda de aplicações não venha crescendo na mesma velocidade que a tecnologia.

Esse fator motiva professores e pesquisadores a identificar novas técnicas e metodologias de ensino para a programação paralela e distribuída e resolver algumas questões como: Como ensinar a programar em um ambiente paralelo? Como pensar em paralelo? Que habilidades são necessárias para capacitar o raciocínio paralelo? Como sistematizar os diferentes tipos de paralelismo para serem ensinados? Ensinar linguagem de programação paralela ou ensinar a desenvolver algoritmos paralelos? Como analisar a eficiência dos algoritmos?

Formalismos de Especificação e Propriedades do Paralelismo

O modelo de computação concorrente envolve propriedades que caracterizam o paralelismo, tais como: decomposição, mapeamento, comunicação e sincronização. Objetiva-se identificar e caracterizar cada uma delas, visando uma formalização da programação paralela. Essa formalização é levada a cabo por técnicas que se baseiam em diagramas (gramática de Grafos e Redes de Petri), em álgebra de processos (CCS e CSP) e em conjuntos (POSETS). Está sendo feito um estudo desses formalismos utilizados para descrever concorrência, para aproximar a teoria da concorrência da resolução de aplicações de alto desempenho (a prática), tendo o estudo de projetos de algoritmos paralelos como interface. Esse estudo objetiva mostrar como esses formalismos descrevem essas propriedades vitais do paralelismo e, a partir disso, transportar esse aprendizado para o campo das linguagens de programação paralelas e para o estudo das metodologias de desenvolvimento dos algoritmos paralelos. Esse estudo é baseado na metodologia de estudo de casos.

Fundamentos da Concorrência

Por outro lado, para o ensino de concorrência é necessário identificar habilidades e conceitos que sejam requisitos, tanto da forma de pensar quanto dos conteúdos a serem desenvolvidos. Dessa forma, identificar-se-á a verdadeira razão da dificuldade do avanço da difusão da programação paralela e distribuída, ou seja, é devido ao fato de os seres humanos serem seres inerentemente sequenciais ou da inexistência da cultura.

Estão sendo estendidos modelos formais de máquinas que representam a computação sequencial, para expressarem algumas das formas de computação paralela, como por exemplo, o uso da Máquina de Turing com várias cabeças e uma fita, para representar ambientes baseados em memória compartilhada. As diferentes cabeças representam os processadores. O protótipo desenvolvido para esse modelo possibilita identificar e estudar conceitos como: desempenho, speedup, eficiência, balanceamento de carga, sincronização, comunicação, entre outros [2]. A simplicidade desse modelo e a facilidade de visualização de seu funcionamento possibilitam a abstração desses conceitos.

Essa projeção da "cultura de alto desempenho", estimula a utilização de formalismos simples e poderosos para explorar a concorrência. São analisadas outras modificações do modelo, Máquina de Turing Paralela, que são extensões do modelo original que se diferem por apresentar mais fitas, unidades de controle e cabeças[3]. A

exploração desses modelos tem a finalidade de verificar a validade do uso de Máquinas de Turing Paralelas como ferramenta para o ensino de programação concorrente. Esse estudo faz-se necessário para que novas gerações sejam estimuladas a apreender programação concorrente.

Conclusão

A Engenharia e a Ciência, cada vez mais, utilizam-se do Processamento de Alto Desempenho, explorando computadores com arquiteturas avançadas. O emprego do paralelismo é essencial para se extrair todo o potencial dessas máquinas e, para isso, o domínio de técnicas e de linguagens de programação paralela são essenciais. Por causa disso, o Grupo de Matemática da Computação e Processamento de Alto Desempenho (GMC-PAD) da UFRGS tem desenvolvido pesquisas com esse intuito.

Referências

- [1] FOSTER, I. T. **Designing and Building Parallel Programs: concepts and tools for parallel software engineering**. Reading: Addison-Wesley, 1995. 379p.
- [2] MARQUEZAN, C. C. et al. Learning Concurrency using Parallel Turing Machine. Porto Alegre: II da UFRGS, 2000. (to appear).
- [3] WIEDERMANN, J. Weak parallel machines: A new class of physically feasible parallel models, In: I.M. Havel, V. Koubek (Eds.), MFCS'92, International Symposium Mathematical Foundations of Computer Science, 17th, Lecture Notes, Springer, 1992 v. 629, pp.95– 111.