

Sistemas Paralelos e Distribuídos

Práticas - Aula 7

Atividade

HPC híbrido utilizando MPI e OpenMP

- Intro
- Considerações de design
- Considerações de execução
- Exemplos
- Métricas

Intro

Sumário

- Ambas técnicas surgiram em 1990
- Utilizar o OpenMP para múltiplas threads
- Utilizar o MPI para múltiplos processos
- Logo no início de sua utilização, programadores notaram que a combinação de ambas era vantajosa em diversos contextos de HPC
- Explorar a hierarquia de paralelismo com múltiplos e múltiplos processadores ao mesmo tempo
- Tornou-se uma “melhor prática” de HPC

Considerações de Design

Sumário

- Como melhor distribuir os dados a serem processados
- Entender possível “race conditions” e implementar mecanismos necessários
- Como melhor alocar os processadores e nós disponíveis
- Considerar afinidades de execuções
- Considerar casos de escalabilidade e definir valores ótimos através de testes preliminares

Considerações de Implementação

Sumário

- Identificar casos de paralelismo
Aritmética pesada (threads) ou alto volume distribuído (comunicação de processos)
- Analisar como melhorar sincronização e comunicação
- Distribuição de carga adequada
- Overhead (cuidado com paralelismo “além do necessário”)
- Considerar thread safety

Exemplos

- Escrever um programa “hello world” que execute em fct-deei-linux e fct-deei-aval

Exemplos

- Escrever um programa que calcule a soma de inteiros de um vetor grande. A execução deve correr em ambos, fct-deei-linux e fct-deei-aval

Métricas

- Usando a soma do vetor, faça a medida das seguintes métricas
 - Speedup
 - Eficiência
 - Tempo de Comunicação
 - Escalabilidade fraca e forte

Exemplos

Representação de possível implementação de multiplicação de matrizes em modelo híbrido

- 1) Inicializar MPI
- 2) Definir matrizes e alocar memória (matrizes, linhas e resultados) no rank 0 e filhos (vetores)
- 3) Envia dados (vetor broadcast para filhos) e particiona matrizes (scatter para filhos)
- 4) Executa a multiplicação com OpenMP
- 5) Faz o gather dos resultados no processo master
- 6) Imprime resultado/Verifica
- 7) Liberta memória
- 8) Finalizar MPI

FIM