

Universidade Federal Rural do Rio de Janeiro

Instituto Multidisciplinar – DCC

Curso de Ciência da Computação

**LABORATÓRIO DE SISTEMAS OPERACIONAIS**

Escalonador de Processos baseado em Afinidade de Memória

Discentes: Miguel Mendes, Raíza Santana e Válber Laux

1. Objetivo do trabalho:

O escalonador de processos de um processador é o componente responsável pela definição da ordem de execução de processos na CPU. Essa organização pode influenciar muito no desempenho dos processos, uma vez que existem processos que necessitam de muito tempo de execução em CPU (conhecidos na literatura como *CPU-bound*) e outros que ficam longos períodos bloqueados aguardando recursos, como conteúdo da memória principal (conhecidos como *I/O-bound*). No entanto, outros fatores podem influenciar no desempenho dos processos. A afinidade de memória entre processos tem sido bastante explorada, especialmente com o advento dos processadores *multicore*. A ideia geral é de que processos afins compartilham dados de memória. Considerando que carregar dados da memória principal é uma tarefa custosa quando comparada a ler dados de registradores ou memórias *cache* (pequenas unidades de memória de rápido acesso e fisicamente próximas do processador) e que dados já carregados, atualmente na *cache*, podem ser compartilhados por esses processos afins, podemos melhorar o desempenho. Quando processos não-afins são escalonados para um mesmo núcleo, podem gerar um efeito de apagar os dados já carregados pelo outro processo com intuito de carregar os seus, mas que logo em seguida serão novamente sobrescritos pelos dados do concorrente, e assim sucessivamente, gerando altas taxas de *cache miss* e consequente degradação do desempenho.

1. Metodologia

Dado esse comportamento, o presente trabalho se propõe a desenvolver um protótipo de escalonador de processos que considera a relação de descendência entre processos como influência no escalonamento. Os resultados serão comparados com a implementação clássica do algoritmo de escalonamento FIFO (*First in, first out*  - primeiro a entrar é o primeiro a sair), que delega o próximo processo da fila ao primeiro processador disponível.

Consideramos para esse trabalho a seguinte arquitetura de processador:

Processador 2

Processador 1

Cache P1

Cache P2

Memória Principal

Figura 1. Arquitetura considerada para os experimentos

Como podemos ver na Figura 1, cada processador possui sua memória *cache*, compartilhando a memória principal entre eles.

Consideraremos como entrada para o algoritmo uma instância que conta com os seguintes dados:

- Quantidade de processos;

Para cada processo:

* Identificador do processo;
* Processo pai
* Tempo estimado de execução
* Tempo real
* Status, que pode ser: 0 - pronto; 1 - bloqueado; 2 - executando; 3 - terminado
* Unidades de tempo antes da entrada do processo na fila em relação ao primeiro processo
* Ciclos de CPU

Um exemplo de instância para 5 processos é exibido abaixo:

> processos.ins

>

> 5

> 1 0 5 0 0 1

> 2 1 2 0 0 3

> 3 0 3 0 0 4

> 4 2 19 0 0 10

> 5 1 2 0 0 13

Foram observados, ao final da execução, os seguintes dados relativos ao desempenho do escalonados:

1. Experimentos

Foi gerado um conjunto de 10 instâncias, geradas aleatoriamente, que foram submetidas a ambos os escalonadores: o proposto pelo grupo, baseado em afinidade de processos, e o baseado na técnica FIFO. Os resultados são apresentados na tabela abaixo. Podemos ver, na primeira coluna...

1. Conclusões

A partir dos resultados apresentados na seção anterior, podemos concluir que a proposta apresentada nesse trabalho indica uma boa proposta de métrica a ser utilizada no escalonamento de processos, quando comparada a técnicas como a do FIFO. O resultado, como também já apontado por outros trabalhos da literatura, é um indicativo de que a influência da memória *cache* e da afinidade de processos é uma peça chave a ser considerada no escalonamento de processos. Vale lembrar que, por se tratar de um estudo teórico, com instâncias que trazem dados normalmente desconhecidos na prática, os resultados não são diretamente aplicáveis à prática, mas indicam bons caminhos para a otimização no escalonamento de processos.

1. Referências

POR MIM NEM VAI TER RISOS