

Pequeno Escalonador por prioridades e créditos

Sistemas Operacionais (ACH2044)

Norton Trevisan Roman

Sistemas de Informação

Universidade de São Paulo

Lucas de Lyra Monteiro (15471435)

São Paulo

2024

[illegible]

Analisando os dados acima é possível perceber que quanto maior o valor de n_com , menor a média de trocas de contexto e maior a média de instruções por quantum. Contudo, somente com estes fatos não se pode concluir que quanto maior o n_com melhor o nosso sistema é.

Ao averiguar mais profundamente percebe-se que temos faixas de valores em que as nossas métricas não se alteram, por exemplo, de 14 a 20 instruções por quantum manteve-se 3,3 trocas de contexto e 4,08 instruções por quantum. Este fato já nos dá indícios que aumentar o número de comandos de 14 para 20, pelo menos no caso deste sistema, não cause benefícios.

Ademais, é interessante avaliar a média de instruções comparado quanto ao número total de instruções disponíveis num quantum. Por exemplo, um valor de $n_com=21$ nos dá o maior valor de instrução por quantum (4,21), entretanto, este valor de 4,21 equivale a somente 20% do máximo de instruções que poderíamos fazer por quantum, enquanto isso com $n_com=3$ temos uma utilização de aproximadamente 78% do número máximo de instruções.

Outras estatísticas interessantes são o número de instruções de entrada e saída (23) e o número de operações COM (71). Infelizmente não consegui achar algum dado de entrada e saída para determinar se o sistema é CPU-Bound ou I/O-bound, mas é importante avaliar isto.

Além disso, não foi especificado o objetivo do sistema descrito, desta forma, fica-se prejudicada uma análise mais assertiva do quantum mais adequado. um sistema de computador pessoal (PC) precisa de um quantum pequeno para agradar o usuário médio, já um servidor ou uma máquina que processe arquivos em batch pode alocar um quantum maior para seus processos.

Assumindo que a operação COM seja consideravelmente gastosa* (i.e, gaste muitos ciclos de clock) e também avaliando o disposto acima podemos chegar em algumas conclusões:

1. Caso o sistema seja pensado para um computador pessoal, como Windows, Ubuntu ou MacOS, é interessante que o mesmo tenha um quantum baixo, de 3 a 7, garantindo que o usuário tenha uma sensação de que os processos estão todos sendo executados de forma síncrona, mesmo que tenhamos troca de contextos mais custosas.
2. Caso o nosso sistema seja um servidor robusto, o melhor seria adotar valores altos, como 21 comandos por quantum, minimizando assim o overhead das trocas de contexto e maximizando as instruções por quantum.

*Um computador atual consegue performar milhões de comandos em curtíssimos espaços de tempo, tendo um time-slice (quantum) entre 10 e 100 milissegundos, normalmente. Então estou supondo que COM é um comando que gasta **muito** tempo (ou que temos um computador muito ruim). Caso contrário, a análise acima faria pouco sentido visto a diferença de ordem de grandeza de duas dezenas de comandos para milhões de comandos.

