

Teste de Escalonabilidade (Escalonamento) – Análise por Tempo de Resposta

Dr. Osmar Marchi dos Santos

Análise por Tempo de Resposta

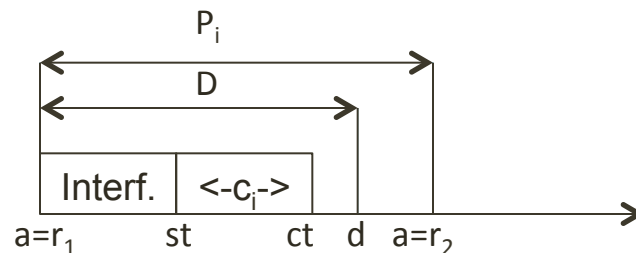
- Análise de escalonamento exata (tanto suficiente como necessária)
 - Do inglês “Response-Time Analysis”
- Utilizada para modelos de escalonamento baseado em prioridades estáticos
- Contém um conjunto grande de extensões como, por exemplo:
 - Lidar com dependência (offsets)
 - Compartilhamento de dados (bloqueio)
 - Comunicação

Conceitos básicos

- Baseada na noção de *instante crítico*
 - Momento em que todas as tarefas do sistema encontram-se na fila de processos prontos para execução
- É pior momento (maior sobrecarga) para a execução das tarefas do sistema
 - Interferência decorrente de tarefas de maior ou igual prioridade sobre as outras tarefas de menor ou igual prioridade
- Caso as tarefas sejam escalonáveis nesse momento, elas serão escalonáveis em qualquer momento

Conceitos básicos

- Utiliza o conceito de tempo de resposta máximo de uma tarefa (*response time*)
 - Tempo da chegada da tarefa até o término de sua execução
 - Considera a interferência de tarefas de maior ou igual prioridade sobre a tarefa sendo analisada



Análise por Tempo de Resposta

- Para calcular o tempo de resposta de uma tarefa i , temos a seguinte equação:

- $$R_i = C_i + \sum_{j \in hp(i)} I_j$$

- Onde:

- $hp(i)$ são as tarefas de maior prioridade que i
- I_j é a interferência que a tarefa i pode sofrer

- $$I_j = \left\lceil R_i / P_j \right\rceil * C_j$$

- Onde:

- R_i é a janela de tempo em análise
- $\left\lceil R_i / P_j \right\rceil$ representa o número de liberações da tarefa j na janela de tempo

Análise por Tempo de Resposta

- A expressão pode ser reescrita como:

- $$R_i = C_i + \sum_{j \in hp(i)} \left\lceil R_i / P_j \right\rceil * C_j$$

- Como R_i aparece em ambos os lados da expressão, é necessário utilizar uma solução iterativa:

- $$R_i^{n+1} = C_i + \sum_{j \in hp(i)} \left\lceil R_i^n / P_j \right\rceil * C_j$$

- $$R_i^0 = C_i$$

- O método converge quando a tarefa é escalonável

Exercício

- Considerando o seguinte sistema, veja se ele é escolanável ou não:

Tarefa	$P_i=D_i$	C_i	Prio. p_i
A	50	5	1
B	70	10	2
C	80	20	3
D	150	20	4
E	150	20	5

Exercício

- Implementar usando a linguagem C:
 - Análise de escalonamento baseada em utilização (aula passada)
 - Análise de escalonamento por tempo de resposta
- Considerar um modelo de tarefas com tarefas independente e deadline menor ou igual ao período das tarefas
 - Alocação de Prioridades utilizando *Deadline Monotonic*
- Considerar que as tarefas são adicionadas a um vetor e seus valores temporais (modelo de tarefas do sistema) são gerados de forma randômica
 - Funções `srand()` e `rand()`
- Abstração: como representar as tarefas e suas informações em um programa C?