

# EMB5633 - Sistemas de Tempo Real

## Análise de Escalonabilidade em Uniprocessadores

Prof. Giovani Gracioli

2020/02

### 1 Descrição

Existem diferentes tipos de testes de escalonabilidade, dependendo do tipo do escalonador (se estático ou dinâmico) e também do tipo do teste (suficiente, exato ou necessário). Mesmo o mesmo escalonador, pode apresentar diferentes testes de escalonabilidade, com diferentes desempenhos (i.e., taxa de escalonabilidade).

Neste trabalho, cada grupo deverá implementar 3 testes de escalonabilidade para o Rate Monotonic (RM) e comparar seus respectivos desempenhos. Os testes de escalonabilidade são: teste suficiente proposto por Liu e Layland em 1970 (disponível nos slides), teste de análise do tempo de resposta (RTA - response time analysis, disponível nos slides) e o hyperbolic bound function (disponível no livro do Buttazzo e complementado neste documento <https://www.di.univr.it/documenti/OccorrenzaIns/matdid/matdid987751.pdf>).

O trabalho consiste em implementar os 3 testes usando uma linguagem de programação de sua preferência (python, C++, R, etc) e compará-los em termos de desempenho, ou seja, quantidade de conjuntos de tarefas escalonáveis. Para isso, deve-se gerar conjuntos de tarefas aleatórios para cada utilização máxima do sistema, variando de 0.1 (10% de utilização) a 1 (100% de utilização) com aumento de 0.1 a cada utilização sendo testada. Assim, as utilizações que serão testadas são: 0.1, 0.2, 0.3, ..., até 1.

Para cada utilização máxima do sistema, deve-se gerar 100 conjuntos de tarefas de forma aleatória e, para cada conjunto de tarefas, aplicar os 3 testes de escalonabilidade, variando-se os parâmetros de geração para cada tarefa (utilização, período e pior tempo de execução).

Os seguintes parâmetros devem ser usados para a geração dos conjuntos de tarefas, sempre usando uma distribuição uniforme.

- Light utilization: [0.0001, 0.01]
- Middle utilization: [0.001, 0.09]
- Heavy utilization: [0.09, 0.1]
- Light period: [3, 33] (em ms)
- Moderate period: [10, 100] (em ms)
- Long period: [50, 250] (em ms)

Com base na utilização e períodos gerados de forma aleatório, tem-se o pior tempo de execução de cada tarefa dentro do conjunto de tarefas. Deve-se combinar as 3 variações de utilização com as 3 variações de período. Por exemplo, light utilization com long period ou middle utilization com light period, e assim por diante.

O resultado deve ser a taxa de escalonabilidade de cada teste para cada utilização. Por exemplo, considerando uma utilização do sistema de 0.8 e 100 conjunto de tarefas. Se o teste de escalonabilidade Liu e Layland conseguiu escalonar 75 dos 100 conjunto de tarefas, ele terá uma taxa de escalonabilidade de 0.75. Se o teste RTA conseguiu escalonar 90 dos 100 conjunto de tarefas, terá uma taxa de 0.9. Já se o teste hyperbolic bound function conseguiu escalonar 100 dos 100 conjunto de tarefas, terá uma taxa de 1.

A Figura 1 abaixo exemplifica o resultado esperado, mostrando a taxa de escalonabilidade (eixo y) para cada um dos testes de escalonabilidade (linhas) e variando a utilização (eixo x).

Com base nos resultados alcançados aponte qual é o melhor teste de escalonabilidade e para quais cenários.

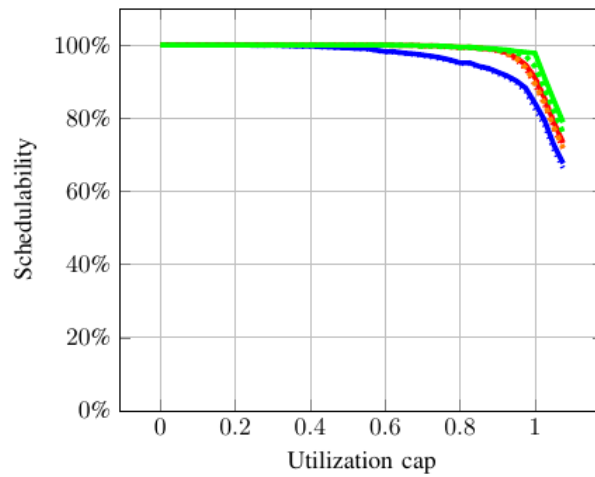


Figure 1: Exemplo de gráfico mostrando a taxa de escalonabilidade de cada um dos testes de escalonabilidade.

## 2 Entrega do Trabalho

Este trabalho deve ser entregue pelo Moodle até a data limite estipulada pela tarefa do Moodle..

Em um único arquivo .zip, envie o código implementado para gerar os conjuntos de tarefas e os testes de escalonabilidade. Anexe um arquivo README explicando como executar o código. Submete também um relatório explicando o código implementando e discutindo os gráficos comparativos gerados.