

# Algoritmos de Escalonamento III

Eduardo Ferreira dos Santos

Ciência da Computação  
Centro Universitário de Brasília – UniCEUB

Abril, 2017

## Sumário

1 Escalonamento Deadline Monotônico

2 Classificação

# 1 Escalonamento Deadline Monotônico

## 2 Classificação

# Conceitos

- Extensão do algoritmo *Rate Monotonic*;
- O deadline relativo igual ao período da tarefa pode ser bastante restritivo.

# Premissas

- 1 Tarefas **periódicas** e **independentes**;
- 2 O tempo de computação de cada tarefa  $C_i$  é conhecido e constante (*Worst Computation Time*);
- 3 O tempo de chaveamento entre as tarefas é considerado **nulo**;
- 4 Deadline relativo menor ou igual ao período da tarefa:  $D_i \leq P_i$

# Atribuição de prioridades no DM

- Ordenação baseada nos *deadlines* relativos das tarefas:
  - Atribuição na ordem inversa do *deadline* relativo;
  - *Deadlines* menores possuem menor prioridade.
- Prioridades fixas;
- Escalonamento *estático* e *online*;
- Também é um algoritmo ótimo na sua classe de problemas.

## Exemplo

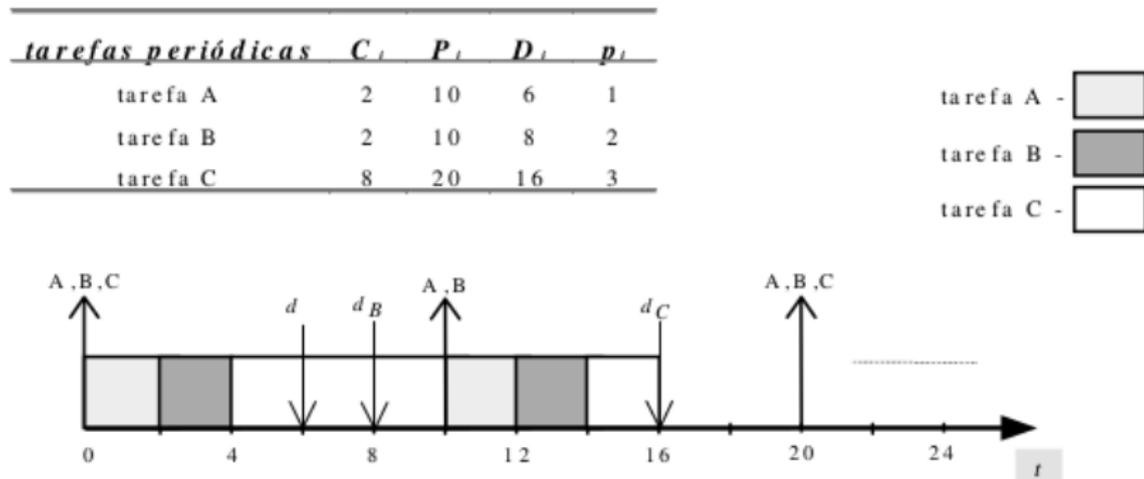


Figura 1.1: Exemplo de escalonamento DM [FARINES and MELO, 2000]

- 1 Escalonamento Deadline Monotônico
- 2 Classificação

# Definições [Chagas, 2016]

**Escalonamento** Ordenar tarefas na fila de pronto;

**Escala** Ordem de ocupação do processador pelas tarefas disponíveis na fila de pronto;

**Escalonador** Programa responsável pela gestão do processador em tempo de execução.

# Estados dos processos

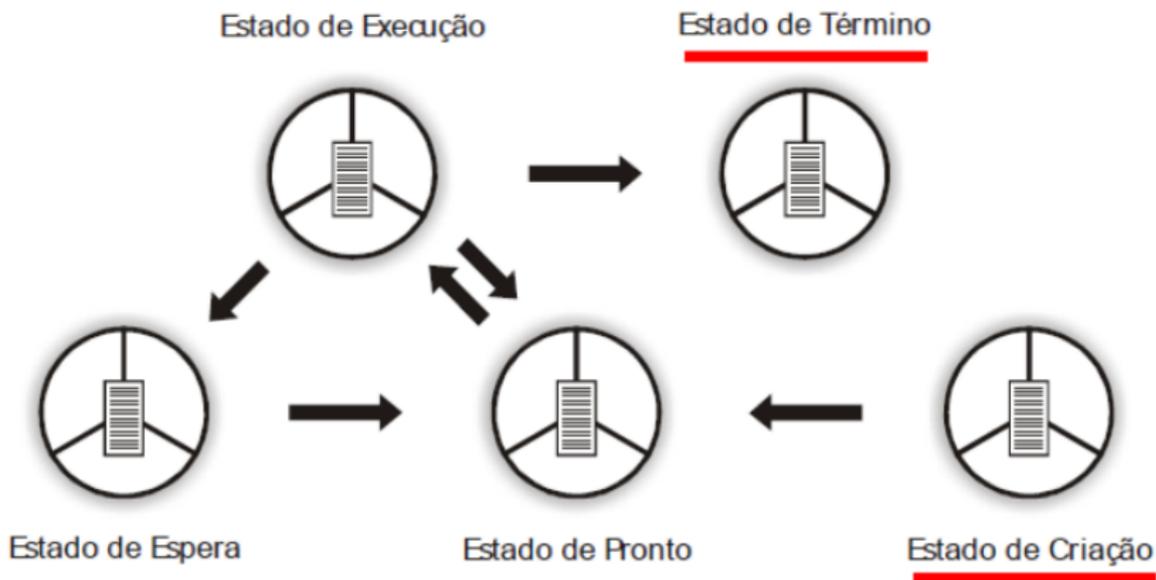


Figura 2.1: Estados dos processos [Chagas, 2016]



# Definição

- Escalonar: Ordenar as tarefas na fila de pronto.

# Definição

- **Escalonar:** Ordenar as tarefas na fila de pronto.
- Os algoritmos de escalonamento podem ser [FARINES and MELO, 2000]:

**Preemptivos** Tarefas podem ser interrompidas em tempo de execução;

**Não preemptivos** Tarefas não podem ser interrompidas por outras mais prioritárias;

**Estáticos** Escalonamento calculado com base em parâmetros fixos atribuídos às tarefas;

**Dinâmicos** Baseados em parâmetros que mudam em tempo de execução.

# Classificação

- Em relação aos parâmetros enviados para as tarefas, os algoritmos podem ser:
  - on-line A escala é produzida em tempo de execução;
  - off-line A escala é produzida em tempo de projeto.
- Os problemas de escalonamento de tempo real podem ser reduzidos a uma **solução polinomial** (NP-Completo)

# Carga computacional [Chagas, 2016]

- **Definição:** Somatório dos tempos de computação na fila de pronto.

## Carga Estática (Limitada)

- Todas as tarefas são bem conhecidas em **tempo de projeto**;
- Modeladas através de tarefas **periódicas e esporádicas**.

## Carga Dinâmica (Ilimitada)

- Características de chegada da tarefa não pode ser antecipada;
- Modeladas através de **tarefas aperiódicas**.

# Abordagens

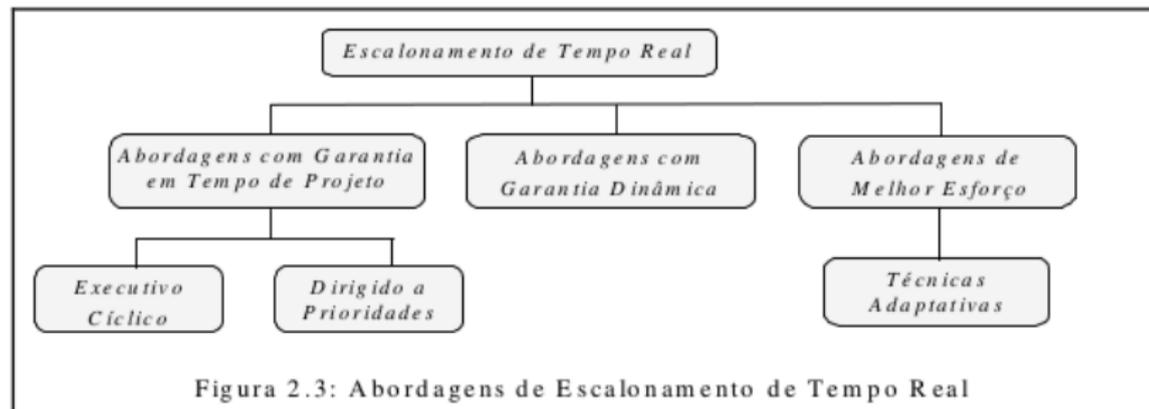


Figura 2.3: Abordagens de escalonamento em tempo real  
[FARINES and MELO, 2000]

# Abordagens de escalonamento [Chagas, 2016]

## Garantia em tempo de projeto I

- Carga computacional dinâmica;
- Teste de escalonabilidade em tempo de projeto.

## Garantia em tempo de execução II

- Carga computacional dinâmica;
- Sistemas críticos que operam em ambientes não determinísticos.

## Abordagens de melhor esforço III

- Teste de escalonabilidade em tempo de execução;
- Não existe previsão de pior caso e não consegue prever recursos para todas as situações de carga.

# Escalonamento estático [Chagas, 2016]

- Escalonamento estático ou executivo cíclico;
- A escala é definida durante a fase do projeto (escalonamento *offline*);
- Tempo de processador atribuído a cada tarefa;
- Garantia de escalonabilidade fornecida pela inspeção da lista de escalonamento (*deadline* da tarefa).

# Dirigido a prioridades

- Dirigidos de acordo com suas prioridades em tempo de execução;
- Prioridades fixas: RM ou DM;
- Prioridades dinâmicas;
- Preemptivos ou não preemptivos.

# Teste de escalabilidade

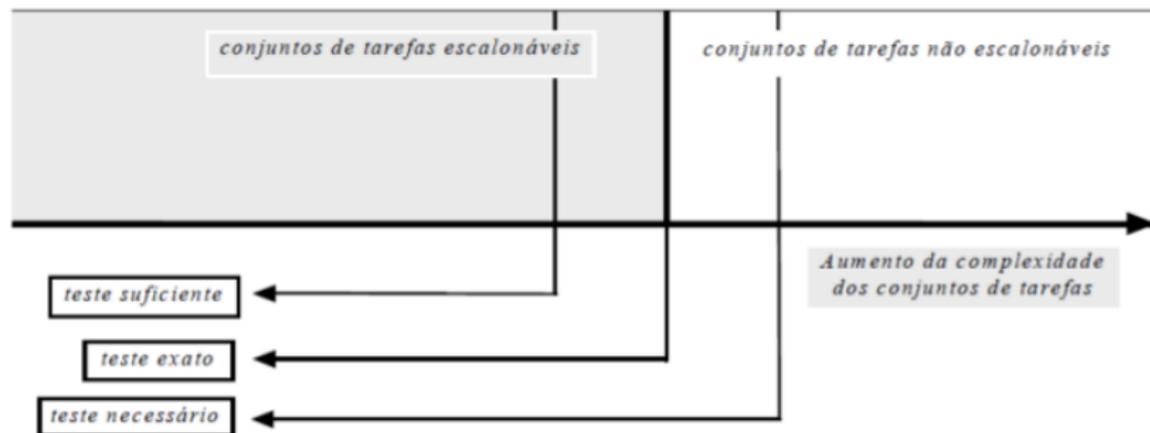


Figura 2.4: Teste de escalabilidade real [FARINES and MELO, 2000]

-  Chagas, F. (2016).  
Notas de aula do Prof. Fernando Chagas.
-  FARINES, J. M. and MELO, R. (2000).  
*Sistemas de Tempo Real*, volume 1.  
IME-USP.

**OBRIGADO!!!**  
**PERGUNTAS???**