

---

## Sistemas de Tempo Real:

### Prioridades com Teste de Escalonabilidade

Rômulo Silva de Oliveira  
Departamento de Automação e Sistemas – DAS – UFSC

romulo@das.ufsc.br  
<http://www.das.ufsc.br/~romulo>  
Junho/2011

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 1

---

### Prioridades + Teste de Escalonabilidade

- Cada tarefa recebe uma prioridade
- Escalonamento em geral é preemptivo
- Teste realizado antes da execução determina escalonabilidade
  - Teste considera como são as tarefas (modelo de tarefas)
    - Periódica, esporádica,  $D <= P$ , bloqueios, etc
  - Teste considera forma como prioridades são atribuídas
  - Validade do teste é demonstrada como teorema
  - Complexidade do teste depende do modelo de tarefas
- Na execução:
  - Escalonador dispara as tarefas conforme as prioridades

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 2

---

### Prioridade com Teste de Escalonabilidade

- **Prioridades Variáveis**

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 3

---

### Prioridades Variáveis – Introdução

- Cada tarefa recebe uma prioridade que varia ao longo do tempo
- Prioridade leva em conta informações relativas à execução
- Diferentes jobs da mesma tarefa podem receber prioridades diferentes
- Cada Job em particular pode receber prioridade fixa ou variável
  
- A escala de execução só é conhecida durante a execução
  
- Necessário Teste de Escalonabilidade
  - Para saber antes se todos os deadlines estão garantidos ou não

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 4

---

### Prioridades Variáveis - Introdução

- **EDF – Earliest Deadline First**
  - Inversamente proporcional ao deadline absoluto
  - Ótimo em relação aos critérios de prioridades variáveis
- **LSF (LST ou LLF) – Least Slack First**
  - Inversamente proporcional ao tempo livre (*laxity* ou *slack*)
  - Ótimo em relação aos critérios de prioridades variáveis
  - Overhead maior que EDF
- **FCFS – First Come First Served**
  - Inversamente proporcional ao tempo de espera por serviço
  - Não é ótimo com respeito ao cumprimento de deadlines
  
- **EDF é o mais usado**

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 5

---

### Prioridades Variáveis – LSF

- **Least Slack First - LSF**
- Quanto menos tempo livre (*slack*), maior a prioridade
- LSF é ótimo quando EDF for ótimo
- Prioridade das tarefas na fila de aptos aumenta com passar do tempo
- Prioridade da tarefa em execução mantém-se constante
  - Gera número maior de chaveamento de contextos que EDF
  - Maior overhead
  
- **Não apresenta vantagens face a EDF**

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 6

## Prioridade com Teste de Escalonabilidade

- Prioridades variáveis
- **Prioridades fixas**

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 7

## Prioridades Fixas – Introdução

- Aplicação composta por tarefas (processos)
- Estados de uma tarefa:
  - Liberada (pronta para executar, apta, ready)
  - Executando (running)
  - Suspensa esperando pela próxima ativação
  - Outros estados serão acrescentados mais adiante
- Em geral escalonamento é preemptivo
- Tarefas possuem **prioridade fixa** definida em projeto
- Garantia exige
  - Tarefas periódicas ou esporádicas
  - Tempo máximo de computação conhecido
  - Teste de escalonabilidade apropriado

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 8

## Prioridades Fixas – Introdução

- Rate Monotonic – RM (Taxa Monotônica)
  - Prioridade mais alta para a tarefa com período menor
  - Prioridade fixa
- Deadline Monotonic – DM (Deadline Monotônico)
  - Prioridade mais alta para a tarefa com deadline relativo menor
  - Prioridade fixa
  - Igual ao RM quando  $D = P$
- Importância
  - Prioridade mais alta para a tarefa mais importante da aplicação
  - Prioridade fixa
- Outras

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 9

## Prioridade Fixa – Teste para RM

- RM – Rate Monotonic (Taxa Monotônica)
- Quanto menor o período, mais alta a prioridade
- Ótimo quando
  - Tarefas são periódicas
  - Deadline é sempre igual ao período
- Exemplo:
 

| Tarefas     | T1    | T2    | T3    |
|-------------|-------|-------|-------|
| Períodos    | P1=30 | P2=40 | P3=50 |
| Prioridades | p1=1  | p2=2  | p3=3  |
- Cuidado!
  - Número menor indica prioridade maior
  - Muitas vezes é o contrário

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 10

## Prioridade com Teste de Escalonabilidade

- Prioridades variáveis
- Prioridades fixas
- **EDF versus Prioridade Fixa**

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 11

## Exemplo escalonabilidade: EDF versus RM

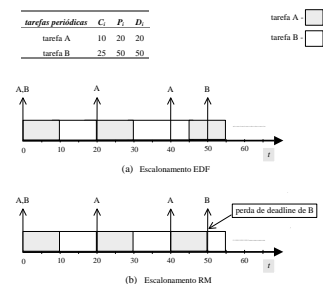
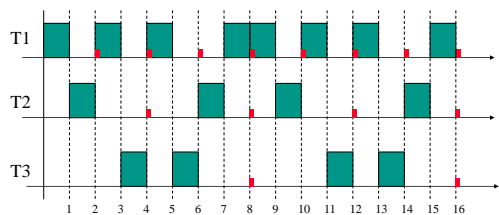


Figura 2.6: Escalas produzidas pelo (a) EDF e (b) RM

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 12

### Exemplo sobrecarga: EDF versus RM

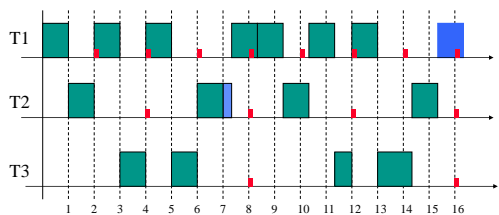
- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Execução normal com EDF



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 13

### Exemplo sobrecarga: EDF versus RM

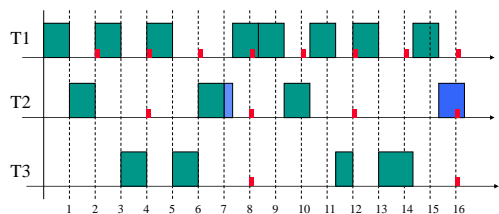
- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Sobrecarga devido a falha do projeto com EDF



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 14

### Exemplo sobrecarga: EDF versus RM

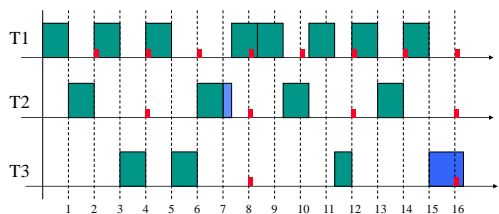
- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Sobrecarga devido a falha do projeto com EDF



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 15

### Exemplo sobrecarga: EDF versus RM

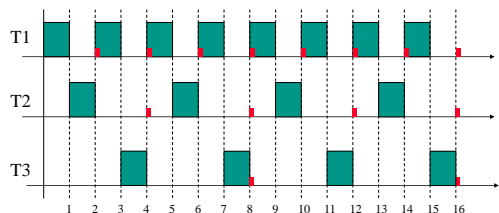
- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Sobrecarga devido a falha do projeto com EDF



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 16

### Exemplo sobrecarga: EDF versus RM

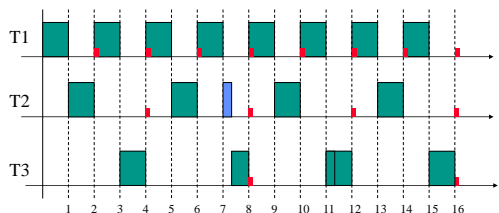
- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Execução normal com RM



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 17

### Exemplo sobrecarga: EDF versus RM

- T1: C1=1 P1=D1=2
- T2: C2=1 P2=D2=4
- T3: C3=2 P3=D3=8
- Sobrecarga devido a falha do projeto com RM



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 18

## Prioridades Variáveis versus Fixas – Comentários

- Implementação é mais complexa do que com prioridade fixa
  - Requer um kernel que aceita prioridades variáveis
  - Precisa recalculer o deadline absoluto para cada job
- *Overhead* de execução pode ser elevado caso seja necessária reordenação dinâmica da fila de aptos (depende do algoritmo)
- Instabilidade face a sobrecargas
  - Não é possível saber antecipadamente quais tarefas vão perder deadline
- Escalonabilidade é superior em EDF do que em prioridade fixa
  - Qualquer sistema escalonável com prioridade fixa também será escalonável com EDF
  - O contrário não é verdadeiro
- Entretanto, prioridade fixa é mais usado na prática

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 19

## Prioridade com Teste de Escalonabilidade

- Prioridades variáveis
- Prioridades fixas
- EDF versus Prioridade Fixa
- Teste de Escalonabilidade Baseado em Utilização

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 20

## Teste baseado na Utilização

- Utilização de uma tarefa:
  - Tempo máximo de computação dividido pelo período
  - Exemplo: T1 tem  $C_1=12$  e  $P_1=50$ , então  $U_1 = 12 / 50 = 0.24$
- Utilização do sistema
  - Somatório da utilização de todas as tarefas
- Dado
  - Um modelo de tarefas
  - Uma política de atribuição de prioridades
- Existe um limiar de utilização para o processador, de tal sorte que:
  - Se a utilização do processador for menor que o limiar
  - Então jamais um deadline será perdido
- Limiar demonstrado como teorema

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 21

## Prioridades Variáveis – Teste para EDF

- Supondo um conjunto de  $n$  tarefas
  - independentes e periódicas
- EDF como política de atribuição de prioridades
- Liu & Layland, 1973

- Se  $D=P$ , sistema é escalonável quando:
  - Permite usar 100% do processador mantendo os deadlines
  - Teste exato
$$\sum_{i=1}^N \left( \frac{C_i}{P_i} \right) \leq 1$$

- Se  $D < P$ , sistema é escalonável quando:
  - Teste suficiente
$$\sum_{i=1}^N \left( \frac{C_i}{D_i} \right) \leq 1$$

- Para  $D$  arbitrário, sistema é escalonável quando:
  - Teste suficiente
$$\sum_{i=1}^N \left( \frac{C_i}{\min(D_i, P_i)} \right) \leq 1$$

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 22

## Prioridades Variáveis – Teste para EDF

- Existe grande variedade de testes de escalonabilidade para EDF
- Testes mais complexos
  - São menos pessimistas
  - Porém requerem esforço computacional maior
- EDF
  - As prioridades de todas as tarefas aptas e em execução aumentam
  - de igual modo com o passar do tempo

- Tarefa possui prioridade variável
- Mas um job em particular possui prioridade fixa

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 23

## Prioridade Fixa – Teste para RM

- Utilização de uma tarefa:
  - Tempo máximo de computação dividido pelo período  $U_i = C_i / P_i$
  - T1 tem  $C_1=12$  e  $P_1=50$ , então  $U_1 = C_1 / P_1 = 12 / 50 = 0.24$
- Liu & Layland, 1973
- Teste para Rate Monotonic, sistema é escalonável se:

$$\sum_{i=1}^N \left( \frac{C_i}{P_i} \right) \leq N(2^{1/N} - 1)$$

- Para  $N=1$  utilização máxima é 100%
- Para  $N$  grandes utilização máxima tende para 69.3%
- Baseado no conceito de Instante Crítico
- Teste é suficiente mas não necessário

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 24

### Prioridade Fixa – Teste para RM

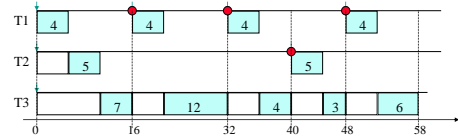
- N Limiar de Utilização

|          |        |
|----------|--------|
| 1        | 100.0% |
| 2        | 82.8%  |
| 3        | 78.0%  |
| 4        | 75.7%  |
| 5        | 74.3%  |
| 10       | 71.8%  |
| infinito | 69.3%  |

### Prioridade Fixa – Teste para RM

- Exemplo:

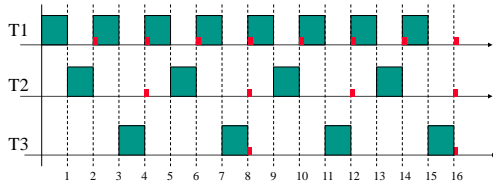
|               |          |          |          |
|---------------|----------|----------|----------|
| T1            | T2       | T3       |          |
| – Períodos    | P1=16    | P2=40    | P3=80    |
| – Computação  | C1=4     | C2=5     | C3=32    |
| – Utilização  | U1=0.250 | U2=0.125 | U3=0.400 |
| – Prioridades | p1=1     | p2=2     | p3=3     |
- Utilização total é 0.775, abaixo do limite 0.780



### Prioridade Fixa – Teste para RM

- Exemplo:

|               |          |          |          |
|---------------|----------|----------|----------|
| T1            | T2       | T3       |          |
| – Períodos    | P1=2     | P2=4     | P3=8     |
| – Computação  | C1=1     | C2=1     | C3=2     |
| – Utilização  | U1=0.500 | U2=0.250 | U3=0.250 |
| – Prioridades | p1=1     | p2=2     | p3=3     |
- Utilização total é 1, acima do limiar 0.780, mas conjunto é escalonável



### Prioridade Fixa – Teste para RM

- Limiar Hiperbólico (Hyperbolic Bound)
- Bini & Buttazzo & Buttazzo, 2001
- Tarefas independentes, P=D, Rate Monotonic
- Se

$$\prod_{i=1}^N (C_i / P_i + 1) \leq 2$$

- Então uma instância de cada tarefa está garantida a cada período
- Suficiente mas não necessário
- Menos pessimista que o teste de Liu & Layland, 1973

### Teste baseado na Utilização – Resumo

- Testes baseados em utilização
  - Não são gerais
  - Não são exatos
  - Mas são rápidos, O(N)
- Exemplo de teste necessário mas não suficiente para este modelo de tarefas

$$\sum_{i=1}^N \left( \frac{C_i}{P_i} \right) \leq 1$$

### Prioridade com Teste de Escalonabilidade

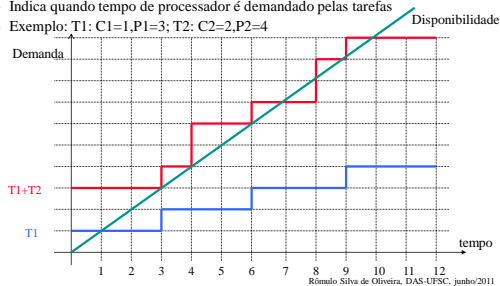
- Prioridades variáveis
- Prioridades fixas
- EDF versus Prioridade Fixa
- Teste de Escalonabilidade Baseado em Utilização
- Teste de Escalonabilidade Baseado na Análise do Tempo de Resposta

### Teste baseado na Análise do Tempo de Resposta

- Limitações da análise baseada em Utilização
  - Não é exata
  - Aplicável apenas a modelos de tarefas muito simples
- Análise baseada em **Tempo de Resposta**
  - Abordagem analítica calcula tempo de resposta no pior caso
  - Tempo de resposta de cada tarefa é comparado com o deadline da tarefa
  - Baseada no conceito de Função Demanda de Tempo

### Teste baseado na Análise do Tempo de Resposta

- Função Demanda de Tempo (Time-Demand Function)
  - Base da análise do tempo de resposta
  - Indica quando tempo de processador é demandado pelas tarefas
  - Exemplo: T1: C1=1,P1=3; T2: C2=2,P2=4



### Prioridade Fixa – Análise do Tempo de Resposta

- Como calcular o tempo de resposta de cada tarefa ?
- Para a tarefa mais prioritária temos  $R_1 = C_1$
- Demais tarefas sofrem **Interferência** das tarefas com prioridade maior
- Neste caso,  $R_i = C_i + I_i$
- Interferência é máxima a partir do **Instante Crítico**
  - Todas as tarefas são liberadas simultaneamente
  - Suposto instante zero na análise

### Prioridade Fixa – Análise do Tempo de Resposta

- Seja  $T_j$  uma tarefa com prioridade maior que  $T_i$
- Quantas vezes  $T_j$  pode acontecer durante a execução de  $T_i$  ?

- Qual a interferência total de  $T_j$  sobre  $T_i$  ?
 
$$\left\lceil \frac{R_i}{P_j} \right\rceil$$
- Qual a interferência total sobre  $T_i$  ?
 
$$\sum_{j \in hp(i)} \left\lceil \frac{R_i}{P_j} \right\rceil \times C_j$$

### Prioridade Fixa – Análise do Tempo de Resposta

- O tempo máximo de resposta de  $T_i$  é  $R_i = C_i + I_i$

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

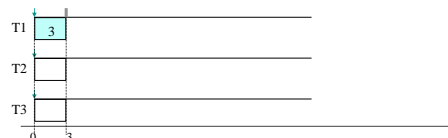
- Equação é recursiva
- Calculada através de iterações sucessivas, até:
  - Tempo de resposta passar do deadline
  - Resultado convergir, iteração  $x+1$  igual a iteração  $x$

$$w_i^{x+1} = C_i + \sum_{j \in hp(i)} \left\lceil \frac{w_i^x}{P_j} \right\rceil \times C_j \quad w_i^0 = C_i$$

### Prioridade Fixa – Análise do Tempo de Resposta

- Exemplo:
 

|               | T1   | T2    | T3    |
|---------------|------|-------|-------|
| – Períodos    | P1=7 | P2=12 | P3=20 |
| – Computação  | C1=3 | C2=3  | C3=5  |
| – Prioridades | p1=1 | p2=2  | p3=3  |
- $R_1 = C_1 = 3$



### Prioridade Fixa – Análise do Tempo de Resposta

- Análise da tarefa T2:

$$w_2^0 = C_2 = 3$$

$$w_2^1 = C_2 + \left\lceil \frac{w_2^0}{P_1} \right\rceil \times C_1 = 3 + \left\lceil \frac{3}{7} \right\rceil \times 3 = 6$$

$$w_2^2 = C_2 + \left\lceil \frac{w_2^1}{P_1} \right\rceil \times C_1 = 3 + \left\lceil \frac{6}{7} \right\rceil \times 3 = 6$$

### Prioridade Fixa – Análise do Tempo de Resposta

- Análise da Tarefa T3:

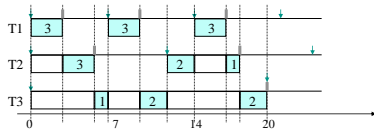
$$w_3^0 = C_3 = 5$$

$$w_3^1 = C_3 + \sum_{j \in hp(3)} \left\lceil \frac{w_3^0}{P_j} \right\rceil \times C_j = 5 + \left\lceil \frac{5}{7} \right\rceil \times 3 + \left\lceil \frac{5}{12} \right\rceil \times 3 = 11$$

$$w_3^2 = 5 + \left\lceil \frac{11}{7} \right\rceil \times 3 + \left\lceil \frac{11}{12} \right\rceil \times 3 = 14 \quad w_3^3 = 5 + \left\lceil \frac{14}{7} \right\rceil \times 3 + \left\lceil \frac{14}{12} \right\rceil \times 3 = 17$$

$$w_3^4 = 5 + \left\lceil \frac{17}{7} \right\rceil \times 3 + \left\lceil \frac{17}{12} \right\rceil \times 3 = 20 \quad w_3^5 = 5 + \left\lceil \frac{20}{7} \right\rceil \times 3 + \left\lceil \frac{20}{12} \right\rceil \times 3 = 20$$

### Prioridade Fixa – Análise do Tempo de Resposta



- Exemplo:
 

|                            | T1   | T2    | T3    |
|----------------------------|------|-------|-------|
| – Períodos                 | P1=7 | P2=12 | P3=20 |
| – Computação               | C1=3 | C2=3  | C3=5  |
| – Prioridades              | p1=1 | p2=2  | p3=3  |
| – Tempo Máximo de Resposta | R1=3 | R2=6  | R3=20 |

### Prioridade Fixa – Análise do Tempo de Resposta

- Exemplo:
 

|               | T1       | T2       | T3       |
|---------------|----------|----------|----------|
| – Períodos    | P1=2     | P2=4     | P3=8     |
| – Computação  | C1=1     | C2=1     | C3=2     |
| – Utilização  | U1=0.500 | U2=0.250 | U3=0.250 |
| – Prioridades | p1=1     | p2=2     | p3=3     |
- Utilização total é 1, acima do limiar 0.780 mas conjunto é escalonável
- Aplicando o cálculo do tempo de resposta temos
- Análise da tarefa T1:
 
$$R1 = C1 = 1$$

### Prioridade Fixa – Análise do Tempo de Resposta

- Exemplo:
 

|               | T1       | T2       | T3       |
|---------------|----------|----------|----------|
| – Períodos    | P1=2     | P2=4     | P3=8     |
| – Computação  | C1=1     | C2=1     | C3=2     |
| – Utilização  | U1=0.500 | U2=0.250 | U3=0.250 |
| – Prioridades | p1=1     | p2=2     | p3=3     |

- Análise da tarefa T2:

$$w_2^0 = C_2 = 1$$

$$w_2^1 = C_2 + \left\lceil \frac{w_2^0}{P_1} \right\rceil \times C_1 = 1 + \left\lceil \frac{1}{7} \right\rceil \times 1 = 2$$

$$w_2^2 = C_2 + \left\lceil \frac{w_2^1}{P_1} \right\rceil \times C_1 = 1 + \left\lceil \frac{2}{7} \right\rceil \times 1 = 2$$

### Prioridade Fixa – Análise do Tempo de Resposta

- Exemplo:
 

|               | T1       | T2       | T3       |
|---------------|----------|----------|----------|
| – Períodos    | P1=2     | P2=4     | P3=8     |
| – Computação  | C1=1     | C2=1     | C3=2     |
| – Utilização  | U1=0.500 | U2=0.250 | U3=0.250 |
| – Prioridades | p1=1     | p2=2     | p3=3     |

- Análise da Tarefa T3:
 
$$w_3^0 = C_3 = 2 \quad w_3^1 = C_3 + \sum_{j \in hp(3)} \left\lceil \frac{w_3^0}{P_j} \right\rceil \times C_j = 2 + \left\lceil \frac{2}{2} \right\rceil \times 1 + \left\lceil \frac{2}{4} \right\rceil \times 1 = 4$$

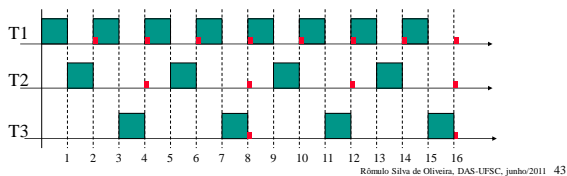
$$w_3^2 = 2 + \left\lceil \frac{4}{2} \right\rceil \times 1 + \left\lceil \frac{4}{4} \right\rceil \times 1 = 5 \quad w_3^3 = 2 + \left\lceil \frac{5}{2} \right\rceil \times 1 + \left\lceil \frac{5}{4} \right\rceil \times 1 = 7$$

$$w_3^4 = 2 + \left\lceil \frac{7}{2} \right\rceil \times 1 + \left\lceil \frac{7}{4} \right\rceil \times 1 = 8 \quad w_3^5 = 2 + \left\lceil \frac{8}{2} \right\rceil \times 1 + \left\lceil \frac{8}{4} \right\rceil \times 1 = 8$$

## Prioridade Fixa – Análise do Tempo de Resposta

- Exemplo:
 

|               | T1       | T2       | T3       |
|---------------|----------|----------|----------|
| – Períodos    | P1=2     | P2=4     | P3=8     |
| – Computação  | C1=1     | C2=1     | C3=2     |
| – Utilização  | U1=0.500 | U2=0.250 | U3=0.250 |
| – Prioridades | p1=1     | p2=2     | p3=3     |
| – Resposta    | R1=1     | R2=2     | R3=8     |
- Utilização total é 1, acima do limiar 0.780, mas conjunto é escalonável



Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 43

## Prioridade Fixa – Análise do Tempo de Resposta

- Teste de escalabilidade **exato** (suficiente e necessário)
- Deadline pode ser **menor** que o período
  - Basta comparar o tempo de resposta com o deadline
- Deadline **maior** que o período exige análise mais complexa
  - Tarefa pode interferir com ela mesma
- Tarefas esporádicas podem ser tratadas como periódicas
  - Intervalo mínimo entre ativações é usado como período
- A forma como prioridades são atribuídas **NÃO** é importante
  - Funciona pois “hp(i)” sempre indica as tarefas mais prioritárias do que a tarefa “i”

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 44

## Prioridade Fixa – Deadline Monotonic

- Quanto menor o deadline, maior a prioridade
- Ótimo quando deadline é **menor ou igual** ao período
- Exemplo:
 

|                            | T1    | T2    | T3    | T4    |
|----------------------------|-------|-------|-------|-------|
| – Tarefas                  |       |       |       |       |
| – Períodos                 | P1=20 | P2=15 | P3=10 | P4=20 |
| – WCET                     | C1=3  | C2=3  | C3=4  | C4=3  |
| – Deadline                 | D1=5  | D2=7  | D3=10 | D4=20 |
| – Prioridades              | p1=1  | p2=2  | p3=3  | p4=4  |
| – Tempo máximo de resposta | R1=3  | R2=6  | R3=10 | R4=20 |
| – Caso fosse RM            | R1=10 | R2=7  | R3=4  | R4=20 |

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 45

## Prioridade Fixa – Deadline Monotonic

- Deadline monotonic (DM) será ótimo se qualquer conjunto de tarefas Q, o qual é escalonável por uma política de atribuição de prioridades W, também for escalonável por DM
- Pode-se provar a optimalidade de DM através da transformação das prioridades de Q (atribuídas por W) até que a ordenação seja aquela do DM
  - Desde que cada passo da transformação preserve a escalonabilidade
- Leung & Whitehead, 1982

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 46

## Prioridade Fixa – Deadline Monotonic

- Sejam  $T_i$  e  $T_j$  duas tarefas de Q, com prioridades adjacentes conforme W, tal que:

$$p_i > p_j \wedge D_i > D_j$$

- Vamos definir a política  $W'$  como sendo idêntica à política W, exceto que as tarefas  $T_i$  e  $T_j$  aparecem invertidas na ordem de prioridades
- Considere a escalonabilidade do conjunto de tarefas Q quando  $W'$  é usado
- Todas as tarefas com prioridades mais altas que  $T_i$  e  $T_j$  não serão afetadas
- Todas as tarefas com prioridades mais baixas que  $T_i$  e  $T_j$  não serão afetadas
  - Continuam recebendo a mesma quantidade de interferência de  $T_i$  e  $T_j$
- A tarefa  $T_j$ , a qual era escalonável com W, agora tem uma prioridade mais alta, sofre menos interferência, portanto também é escalonável quando  $W'$  é usado

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 47

## Prioridade Fixa – Deadline Monotonic

- Resta saber se a tarefa  $T_i$ , cuja prioridade foi reduzida, continua escalonável
- Usando W temos:
  - $R_j \leq D_j$ ,  $D_j < D_i$ ,  $D_i \leq P_i$  e  $D_j \leq P_j$
  - como  $p_i > p_j$  temos  $R_i < R_j$
- A partir do instante crítico, é possível executar uma instância de  $T_i$  e uma de  $T_j$  antes de  $D_j$ :
  - $R_i \leq D_j$  e  $R_j < R_i \leq D_i$
- e antes que qualquer uma das duas chegue novamente
  - $R_j \leq D_j \leq P_j$  e  $R_i \leq D_i \leq P_i$
  - $R_i < R_j \leq P_j$  e  $R_j \leq D_j < D_i \leq P_i$
- A troca das duas prioridades vai simplesmente trocar a ordem na linha do tempo:
  - $R_i' = R_j \leq D_j < D_i$
- Tarefa  $T_i$  continua escalonável depois da troca com  $T_j$

Rômulo Silva de Oliveira, DAS-UFSC, junho/2011 48



### Prioridade Fixa – Deadline Monotonic

---

- Tarefa  $T_i$  continua escalonável depois da troca com  $T_j$
- Agora a política  $W^*$  pode ser transformada em  $W^{**}$ 
  - Escolhe-se mais duas tarefas que não estão de acordo com DM
- No final chegamos na ordem de DM, com um conjunto de tarefas escalonável

### Escalonamento com Teste – Resumo

---

- Cada tarefa recebe uma prioridade fixa
- Teste é desenvolvido para examinar a escalonabilidade
- Dois tipos de análise
- **Análise da Utilização**
  - Utiliza o valor C/P
- **Análise do Tempo de Resposta**
  - Tenta calcular o tempo de resposta no pior caso

### Escalonamento com Teste – Resumo

---

- Prioridades + Teste de Escalonabilidade
  - Prioridades Variáveis com Teste de Escalonabilidade
  - Prioridades Fixas com Teste de Escalonabilidade
- Todos são capazes de garantir deadlines
  - Executivo cíclico é menos flexível porém oferece determinismo de escala
  - EDF oferece escalonabilidade superior em relação à prioridade fixa, porém é caótico em sobrecarga e análise é mais complexa
  - Prioridade fixa permite estender mais facilmente a análise de escalonabilidade