Generalized fixed-priority scheduling with limited preemptions

Reinder J. Bril, Martijn M.H.P. van den Heuvel, Ugur Keskin, and Johan J. Lukkien

Dep. of Mathematics and Computer Science
Group System Architecture and Networking

ECRTS-2012 (Pisa, Italy)
Motivation

- Drawbacks of arbitrary preemptions (FPPS):
  - worst-case memory requirements;
  - cost of context switching (e.g. cache).
- Non-preemptive scheduling (FPNS):
  - resolved by FPNS,
  - at the cost of lower schedulability.
- Alternative refinements of FPPS (FPTS & FPDS):
  - reduced memory costs (compared to FPPS);
  - improved schedulability (compared to FPPS);
  - orthogonal approaches!
Motivation

• **Goal:**
  - combine strength of FPTS and FPDS in a single scheme: FPGS,
  - with the aim to improve efficiency,
  - focus on improvement of the feasibility.
Overview

- Motivation
- Introduction fixed-priority scheduling (FPS)*
  - model
  - FPPS and FPNS
- Existing limited-preemptive algorithms*
  - preemption thresholds (FPTS)
  - deferred preemption (FPDS)
- Novel hybrid algorithms
- Conclusion

* Buttazzo, Bertogna, and Yao, IEEE TII, 2012.
Introduction FPS – model

- Events: implicit
- Tasks (τ):
  - independent, no self-suspension
- characteristics (R+):
  - minimal inter-arrival time (T);
  - computation time (C);
  - deadline (D);
- Scheduling algorithm:
  - fixed-priority (π) & non-idling;
  - [non-] preemptive
- Platform: single CPU
Introduction FPS – FPPS and FPNS

• **FPPS:**
  • highest priority task with work pending executes;
  • a task experiences *interference* from *higher* priority tasks (due to *delays* and *preemptions*).

• **FPNS:**
  • tasks run to completion;
  • the highest priority task is selected to run next;
  • a task experiences:
    - *interference* from *higher* priority tasks (only *delays*);
    - *blocking* from *lower* priority tasks.
Introduction FPS – FPPS and FPNS

• FPNS:
  • blocking: \( B_i = \max(0, \max_{l: \pi_i > \pi_l} C_l) \)

• **Blocking tolerance** (\( \beta_i \)) [1]:
  • the maximum amount of time that a task (\( \tau_i \)) can be blocked without missing its deadline (\( D_i \));
  • depends on scheduling algorithm.

• Neither FPPS dominates FPNS nor vice versa.

Existing limited-preemptive algorithms

- Two orthogonal approaches: FPTS [2, 3] and FPDS

- **FPTS**: *preemption threshold* ($\theta_i \geq \pi_i$)
  - interference: (reduce preemptions)
  - tasks $\tau_h$ with $\pi_h > \theta_i$ can preempt $\tau_i$;
  - blocking: $B_i = \max(0, \max_{l: \theta_l \geq \pi_i, \pi_l} C_l)$

- **Special cases of FPTS**: 
  - **FPPS**: $\theta_i = \pi_i$;
  - **FPNS**: $\theta_i = \pi_1$.

FPTS – preemption thresholds

<table>
<thead>
<tr>
<th></th>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>70</td>
<td>50</td>
<td>20</td>
<td>3</td>
<td>20</td>
<td><strong>55</strong></td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>80</td>
<td>80</td>
<td>20</td>
<td>2</td>
<td>40</td>
<td>75</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>200</td>
<td>100</td>
<td>35</td>
<td>1</td>
<td><strong>115</strong></td>
<td>75</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 30$, $C_2 < \beta_1 < C_3$.

Blocking tolerance $\tau_2$:
- FPPS: $\beta_2^P = 30 < C_3$;
- FPNS: $\beta_2^N = 40 > C_3$.

**Not** scheduable with FPPS

**Not** scheduable with FPNS
FPTS – preemption thresholds

<table>
<thead>
<tr>
<th></th>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
<th>$\theta_i$</th>
<th>$WR_i^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>70</td>
<td>50</td>
<td>20</td>
<td>3</td>
<td>20</td>
<td></td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>80</td>
<td>80</td>
<td>20</td>
<td>2</td>
<td>40</td>
<td>75</td>
<td>3</td>
<td>75</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>200</td>
<td>100</td>
<td>35</td>
<td>1</td>
<td>115</td>
<td>75</td>
<td>2</td>
<td>95</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 30$, $C_2 < \beta_1 < C_3$.

Blocking tolerance $\tau_2$:
- FPPS: $\beta_2^P = 30 < C_3$;
- FPNS: $\beta_2^N = 40 > C_3$.

Schedulable with FPTS
Existing limited-preemptive algorithms

- Two orthogonal approaches: FPTS and FPDS [4, 5]

- FPDS: deferred preemption
  - a task is a sequence of $m_i$ non-preemptive sub-tasks;
  - characteristic subtask $\tau_{i,k}$: computation time $C_{i,k}$;
  - tasks $\tau_h$ with $\pi_h > \pi_i$ can only preempt $\tau_i$ at preemption points (between subtasks);
  - blocking: $B_i = \max(0, \max \max_{l\pi_i>\pi_l} C_{l,k})$

- Special case of FPDS:
  - FPNS: $m_i = 1$.

FPDS – deferred preemption

<table>
<thead>
<tr>
<th></th>
<th>$T_i = D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
<th>$WR_i^D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>7</td>
<td>2+2</td>
<td>1</td>
<td>8</td>
<td>6</td>
<td>7</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 3 < C_3$. 

Schedulable with FPDS
FPDS – deferred preemption

<table>
<thead>
<tr>
<th>$T_i = D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
<th>$WR_i^D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>5</td>
<td>2</td>
<td>2</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>7</td>
<td>2+2</td>
<td>1</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 3 < C_3$.

- **FPTS:**
  - $\theta_2 = \pi_2$: FPPS
  - $\theta_2 = \pi_1$: FPNS

- **Conclusion:**
  - Not schedulable with FPTS, hence
  - FPTS does not dominate FPDS
FPDS does not dominate FPTS

• Previous slide:
  • FPTS does not dominate FPDS;

• Without example (space & time reasons):
  • FPDS does not dominate FPTS

• Conclusion:
  • neither FPDS dominates FPTS nor vice versa
Novel hybrid algorithms

<table>
<thead>
<tr>
<th></th>
<th>$m_i = 1$</th>
<th>$m_i \geq 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{i,k} = \pi_i$</td>
<td>FPPS $\leftrightarrow$ FPPS$^+$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FPTS $\leftarrow$ FPTS$^+$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FPNS $\leftarrow$ FPDS</td>
<td></td>
</tr>
</tbody>
</table>

Generalization graph for FPS algorithms
Novel hybrid algorithms – model

- **FPTS⁺** [6, 7]
  - a task is a sequence of $m_i$ sub-tasks;
  - each subtask $\tau_{i,k}$ has a preemption threshold $\theta_{i,k}$;
  - a task has no (longer a) preemption threshold;
  - blocking: $B_i = \max(0, \max_{l: \pi_i > \pi_l} \max_{1 \leq k \leq m_i: \theta_{i,k} \geq \pi_i} C_{l,k})$

- **Special cases for FPPS⁺:**
  - $m_i = 1$: FPTS;
  - $\theta_{i,k} = \pi_i$: FPPS⁺;
  - $\theta_{i,k} = \pi_1$: FPDS.

FPTS⁺ – preemption thresholds

<table>
<thead>
<tr>
<th></th>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>15</td>
<td>15</td>
<td>7+1</td>
<td>2</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>26</td>
<td>17</td>
<td>1+5</td>
<td>1</td>
<td>26</td>
<td>16</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 1$, $C_3 > \beta_1$, $C_2 > \beta_1$.

Blocking tolerance $\beta_1 = 1$ ⇒ **Not** schedulable with FPDS.
FPT$^+$ – preemption thresholds

<table>
<thead>
<tr>
<th>$\tau_1$</th>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR^P_i$</th>
<th>$WR^N_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>15</td>
<td>15</td>
<td>7+1</td>
<td>2</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>26</td>
<td>17</td>
<td>1+5</td>
<td>1</td>
<td>26</td>
<td>16</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
• $\beta_1 = 1$, $C_3 > \beta_1$, $C_2 > \beta_1$.

Blocking tolerance $\tau_2$:
• FPPS: $\beta_2^P = 4 < C_3$;

Blocking tolerance $\beta_2 < C_3 \Rightarrow \textbf{Not} \text{ schedulable with FPT}$.
### FPTS+ – preemption thresholds

<table>
<thead>
<tr>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
<th>$\theta_{i,1}$</th>
<th>$\theta_{i,2}$</th>
<th>$WR_i^{T^+}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td><strong>9</strong></td>
<td>3</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>15</td>
<td>15</td>
<td>7+1</td>
<td>2</td>
<td>10</td>
<td>15</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>26</td>
<td>17</td>
<td>1+5</td>
<td>1</td>
<td><strong>26</strong></td>
<td>16</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

**Blocking tolerance $\tau_1$:**
- $\beta_1 = 1$, $C_3 > \beta_1$, $C_2 > \beta_1$.

**Blocking tolerance $\tau_2$:**
- FPPS: $\beta_2^P = 4 < C_3$;
- FPTS+: $\beta_2^{T^+} = 5$
FPTS+ – preemption thresholds

<table>
<thead>
<tr>
<th></th>
<th>$T_i$</th>
<th>$D_i$</th>
<th>$C_i$</th>
<th>$\pi_i$</th>
<th>$WR_i^P$</th>
<th>$WR_i^N$</th>
<th>$\theta_{i,1}$</th>
<th>$\theta_{i,2}$</th>
<th>$WR_i^{T+}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\tau_1$</td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>9</td>
<td>3</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>$\tau_2$</td>
<td>15</td>
<td>15</td>
<td>7+1</td>
<td>2</td>
<td>10</td>
<td>15</td>
<td>2</td>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>$\tau_3$</td>
<td>26</td>
<td>17</td>
<td>1+5</td>
<td>1</td>
<td>26</td>
<td>16</td>
<td>1</td>
<td>2</td>
<td>17</td>
</tr>
</tbody>
</table>

Blocking tolerance $\tau_1$:
- $\beta_1 = 1$, $C_3 > \beta_1$, $C_2 > \beta_1$.

Blocking tolerance $\tau_2$:
- FPPS: $\beta_2^P = 4 < C_3$;
- FPTS+: $\beta_2^{T+} = 5$
Novel hybrid algorithms

<table>
<thead>
<tr>
<th>$m_i = 1$</th>
<th>$m_i \geq 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_i &gt; 1 \Rightarrow \theta_i = \pi_i$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\theta_{i,k} = \pi_i$</th>
<th>$\theta_{i,k} = \pi_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPPS</td>
<td>FPPS$^+$</td>
</tr>
<tr>
<td>$\uparrow$</td>
<td>$\uparrow$</td>
</tr>
<tr>
<td>FPTS</td>
<td>FPTS$^+$</td>
</tr>
<tr>
<td>$\downarrow$</td>
<td>$\downarrow$</td>
</tr>
<tr>
<td>FPNS</td>
<td>FPDS</td>
</tr>
<tr>
<td>$\leftarrow$</td>
<td>$\leftarrow$</td>
</tr>
<tr>
<td></td>
<td>FPDS$^\wedge$</td>
</tr>
</tbody>
</table>

Generalization graph for FPS algorithms
Novel hybrid algorithms – results

1. Novel scheduling algorithms: FPGS
   • subtasks (similar to FPDS);
   • preemption thresholds for tasks (FPTS) and subtasks;
   • generalizes existing FPS algs.

2. Schedulability analysis for FPGS
   • specializes to all existing FPS algs;

3. Algorithm to maximize schedulability under FPS:
   • given: $T_i, D_i, C_i$, and $\pi_i$;
   • determine: $C_{i,mi}$, $\theta_i$, $\theta_{i,mi}$ (inspired by [8]).


Novel hybrid algorithms – evaluation

10 tasks, 10,000 task sets,

\( T_i \in [100, 10,000] \) (uniform), \( U_i \) by UUnifast \((\Rightarrow C_i)\),

\( D_i \in [0.5(T_i + C_i), T_i] \) (uniform);
Conclusion

• FPGS and existing FPS algorithms:
  • FPGS generalizes all others;
  • analysis of FPGS specializes to all others;
  • FPGS dominates all others.

• Future work:
  • further improvements of schedulability:
    − preemption thresholds for preemption points;
  • context switching cost;
  • ...

ECRTS-2012 (Pisa, Italy)