Work-in-Progress: Preference-Oriented Scheduling in Multiprocessor Real-Time Systems

Qin Xia[1], **Dakai Zhu**[2] and Hakan Aydin[3]

[1]Xian Jiaotong University, China

[2]The University of Texas at San Antonio, USA

[2]George Mason University, USA

2018 RTSS, Nashville, TN

UTSA
Computer Science

Beyond Timeliness: Execution Preferences!

❖ **Classical RT schedulers treat tasks the same**

- **EDF/RMS: execute all tasks at their earliest times**
- **EDL/DP: execute all tasks at their latest times**

❖ **Tasks may prefer to be executed early or late**

- **Fault-tolerant systems w. primary/backup model**
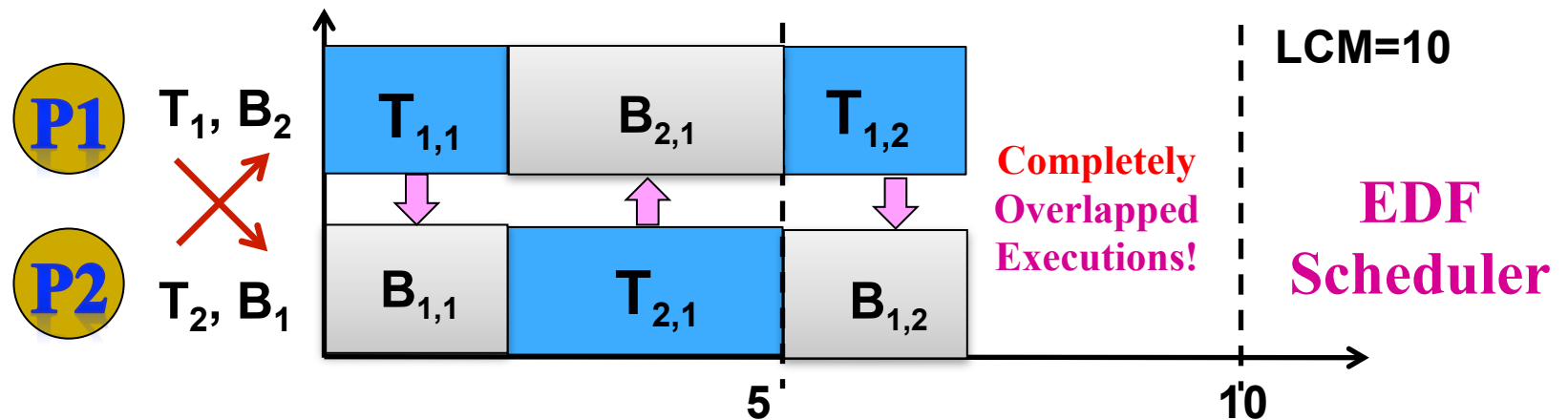
An example of two tasks T_1 (2, 5) , T_2 (3, 10) with their **backups** B_1 and B_2

Beyond Timeliness: Execution Preferences!

- **Classical RT schedulers treat tasks the same**
  - **EDF/RMS: execute all tasks at their earliest times**
  - **EDL/DP: execute all tasks at their latest times**

- **Tasks may prefer to be executed early or late**
  - **Fault-tolerant systems w. primary/backup model**
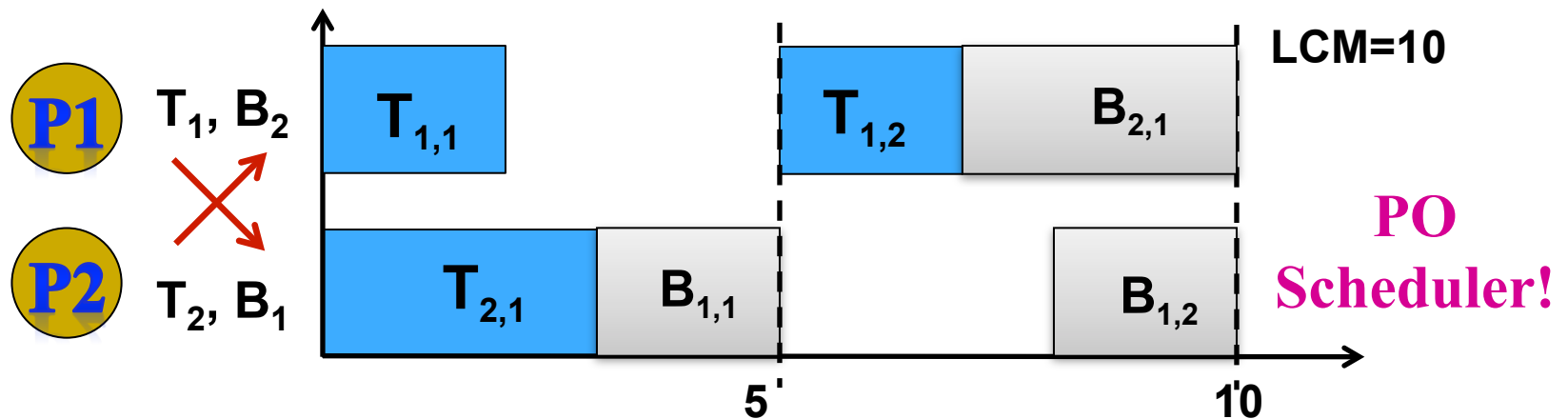
An example of two tasks T_1 (2, 5) , T_2 (3, 10) with their **backups** B_1 and B_2

Models and Prior Work

❖ **Real-time tasks with execution preferences**

- **Set Ψ of periodic task T_i (c_i, p_i): WCET, period**

- **Each task has a preference:** $\qquad \Psi = \Psi_S \cup \Psi_L$

  ✓ **Early tasks:** as soon as possible (ASAP)

  ✓ **Late tasks:** as late as possible (ALAP)

- **Preference value (PV)**

  ✓ **For each task instance**

  ✓ **Early/late tasks: finish/start times**

$$PV_{i,j} = \begin{cases} \dfrac{ft_{\max} - ft}{ft_{\max} - ft_{\min}}, (T_i \in \Psi_S) \\[2mm] \dfrac{st - st_{\min}}{st_{\max} - st_{\min}}, (T_i \in \Psi_L) \end{cases}$$

❖ **Prior work: Preference-Oriented Schedulers**

- **POED: deadline based PO scheduler** [Guo'15]

- **POFP: fixed-priority based PO schedule** [Begam'16]

UTSA Computer Science

PO Scheduling: Multiprocessor Systems

❖ **Several observations for PO partitioning**

- **Each processor: mixture of early/late tasks**

- **Early/late tasks w. harmonic periods → same P**

❖ **POPA (period-aware) partitioning**

- **Tasks w. harmonic periods → pairs (or groups)**

  ✓ **Task pair: one early and one late tasks w. same period**

  ✓ **Task group: balanced early/late task utilizations**

- **Sort task pairs (groups) w. aggregated utilization**

- **Map task pairs (groups) → processors**

  ✓ **WFD or other heuristics**

  ✓ **Map remaining individual tasks at the end**

UTSA
Computer Science