Time-Triggered Mixed-Critical Scheduler [1]

Dario Socci, Peter Poplavko, Saddek Bensalem and Marius Bozga

Verimag - Université Joseph Fourier - Grenoble

Workshop on Mixed Criticality Systems

[1] The research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement no. 288175 [Certainty].

Schedulability
in Dual Critical Systems

- scheduling is a major challenge of Mixed Critical System design
- finite set of jobs in dual critical systems:
  - every job is is classified as critical (HI) or non-critical (LO)
  - every job is labeled with two Worst Case Execution Times:

    LO WCET computed with industrial standard tools (realistic estimation)

    HI WCET computed with Certification Authority tools (very pessimistic estimation)
  - both HI and LO jobs are Hard Real-Time

Schedulability

in Dual Critical Systems

- a set of jobs is correctly scheduled if:

Schedulability
in Dual Critical Systems

- a set of jobs is correctly scheduled if:
  - Condition 1: If all jobs respect their LO WCET, then both HI and LO jobs must meet their deadline.

Schedulability

in Dual Critical Systems

- a set of jobs is correctly scheduled if:
  - Condition 1: If all jobs respect their LO WCET, then both HI and LO jobs must meet their deadline. (VALIDATION)

Schedulability

in Dual Critical Systems

- a set of jobs is correctly scheduled if:
  - Condition 1: If all jobs respect their LO WCET, then both HI and LO jobs must meet their deadline. (VALIDATION)
  - Condition 2: If at least one job's execution time exceeds its LO WCET, then all HI jobs must complete before their deadline, whereas LO jobs may be even dropped.

Schedulability

in Dual Critical Systems

- a set of jobs is correctly scheduled if:
  - Condition 1: If all jobs respect their LO WCET, then both HI and LO jobs must meet their deadline. (VALIDATION)
  - Condition 2: If at least one job's execution time exceeds its LO WCET, then all HI jobs must complete before their deadline, whereas LO jobs may be even dropped. (CERTIFICATION)

Schedulability

in Dual Critical Systems

- a set of jobs is correctly scheduled if:
  - Condition 1: If all jobs respect their LO WCET, then both HI and LO jobs must meet their deadline. (VALIDATION)
  - Condition 2: If at least one job's execution time exceeds its LO WCET, then all HI jobs must complete before their deadline, whereas LO jobs may be even dropped. (CERTIFICATION)
- the scheduling problem is NP-complete

Basic scenario

- an execution where all jobs terminates after executing for $C(LO)$ before a *switch time* t_s and for $C(HI)$ after t_s is called basic scenario
- analyzing basic scenario is enough
- we call a basic scenario LO if no jobs run for $C(HI)$
  $HI - J_s$ if J_s is the first job to execute for more than $C(LO)$

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

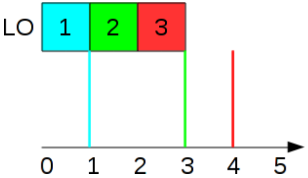
- EDF

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF

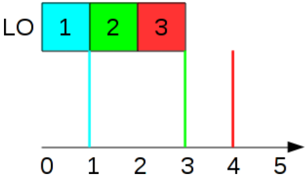


Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



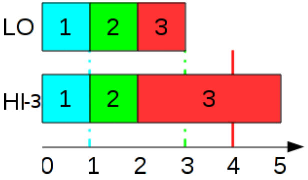
- VERIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



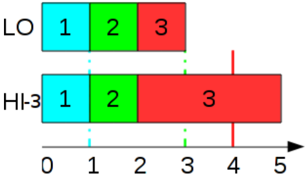
- VERIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



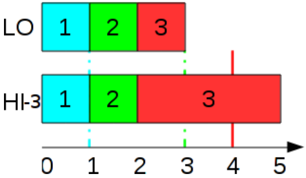
- VERIFIED
- NOT CERTIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- Criticality Monotonic

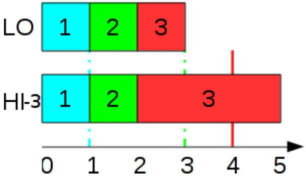
- VERIFIED
- NOT CERTIFIED

Example of scheduling problem

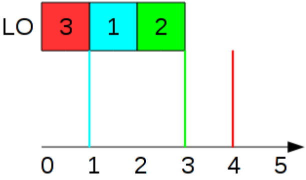
- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- Criticality Monotonic



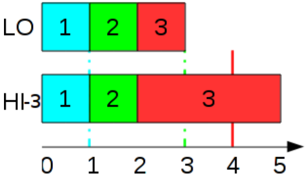
- VERIFIED
- NOT CERTIFIED

Example of scheduling problem

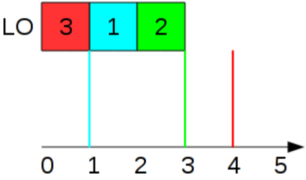
- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- Criticality Monotonic



- VERIFIED
- NOT CERTIFIED

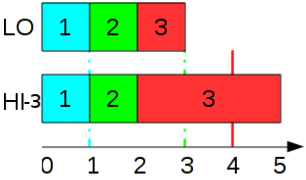
- NOT VERIFIED

Example of scheduling problem

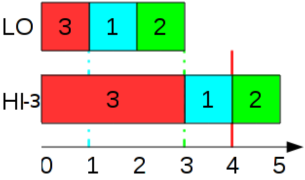
- Let us apply different scheduling to the instance:

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- Criticality Monotonic



- VERIFIED
- NOT CERTIFIED

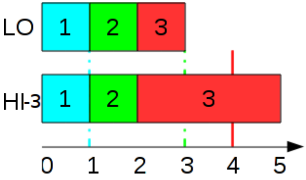
- NOT VERIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

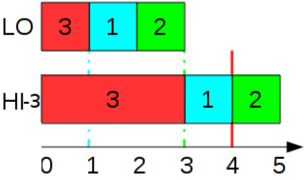
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



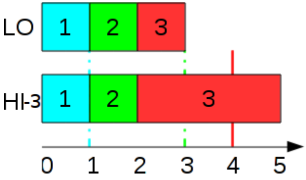
- NOT VERIFIED
- CERTIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

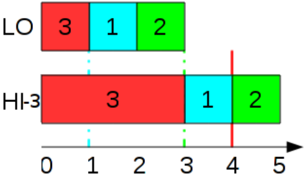
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



- NOT VERIFIED
- CERTIFIED

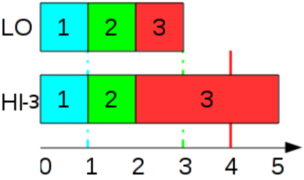
- priority order = 1-3-2

Example of scheduling problem

- Let us apply different scheduling to the instance:

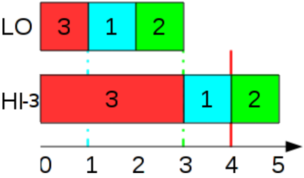
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



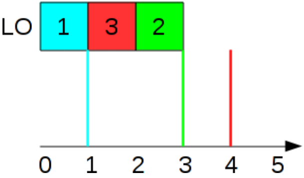
- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



- NOT VERIFIED
- CERTIFIED

- priority order = 1-3-2

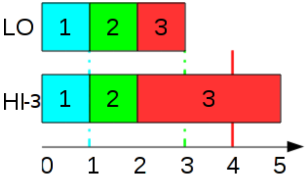


Example of scheduling problem

- Let us apply different scheduling to the instance:

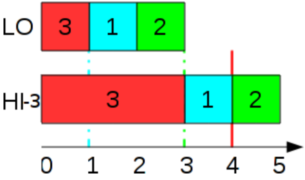
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



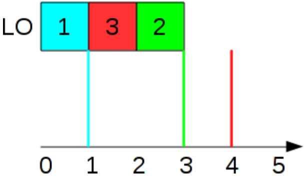
- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



- NOT VERIFIED
- CERTIFIED

- priority order = 1-3-2



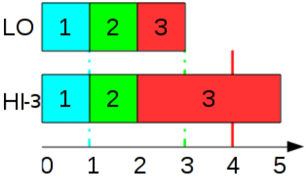
- VERIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

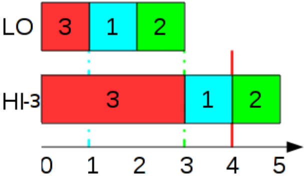
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



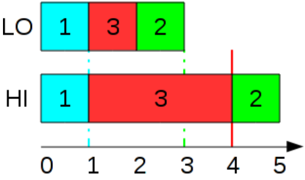
- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



- NOT VERIFIED
- CERTIFIED

- priority order = 1-3-2



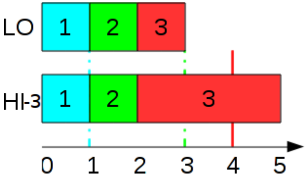
- VERIFIED

Example of scheduling problem

- Let us apply different scheduling to the instance:

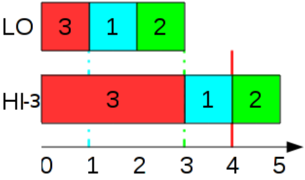
Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3

- EDF



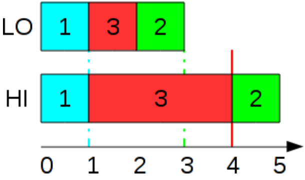
- VERIFIED
- NOT CERTIFIED

- Criticality Monotonic



- NOT VERIFIED
- CERTIFIED

- priority order = 1-3-2



- VERIFIED
- CERTIFIED

Priority based and Time Triggered scheduler approach

- Fixed Priority (FP)
  - fixed priority *per job* table
  - “mode ignorant”

Priority based and Time Triggered scheduler approach

- Fixed Priority (FP)
  - fixed priority *per job* table
  - "mode ignorant"
- Fixed Priority per Mode (FPM)
  - one priority table per criticality mode
  - the knowledge of the current *mode* improves schedulability

Priority based and Time Triggered scheduler approach

- Fixed Priority (FP)
  - fixed priority *per job* table
  - “mode ignorant”
- Fixed Priority per Mode (FPM)
  - one priority table per criticality mode
  - the knowledge of the current *mode* improves schedulability
- Time Triggered (TT) Schedulers
  - the scheduler intervals are precomputed (certification is easier)
  - “mode ignorant”

Priority based and Time Triggered scheduler approach

- Fixed Priority (FP)
  - fixed priority *per job* table
  - "mode ignorant"
- Fixed Priority per Mode (FPM)
  - one priority table per criticality mode
  - the knowledge of the current *mode* improves schedulability

- Time Triggered (TT) Schedulers
  - the scheduler intervals are precomputed (certification is easier)
  - "mode ignorant"
- Single Time Table per Mode (STTM)[1]
  - one TT table per criticality mode

[1] S. Baruah and G. Fohler. "Certification-Cognizant Time-Triggered Scheduling of Mixed-Criticality Systems". In: *Real-Time Systems Symposium (RTSS), 2011 IEEE 32nd*.

Overview

Consider the set of jobs schedulable by the following approaches:

$FP \quad OCBP \quad MCEDF \quad FPM \quad STTM$

Overview

Consider the set of jobs schedulable by the following approaches:

$$FP \overset{[2]}{=} OCBP \qquad MCEDF \qquad FPM \qquad STTM$$

[2] Sanjoy K. Baruah, Haohan Li, and Leen Stougie. "Towards the Design of Certifiable Mixed-criticality Systems". In: *Real-Time and Embedded Technology and Applications Symposium*. RTAS'10.

Overview

Consider the set of jobs schedulable by the following approaches:

$$FP \overset{[2]}{=} OCBP \overset{[3]}{\subsetneq} MCEDF \qquad FPM \qquad STTM$$

[2] Sanjoy K. Baruah, Haohan Li, and Leen Stougie. "Towards the Design of Certifiable Mixed-criticality Systems". In: *Real-Time and Embedded Technology and Applications Symposium*. RTAS'10.

[3] Dario Socci et al. "Mixed Critical Earliest Deadline First". In: *Euromicro Conf. on Real-Time Systems*. ECRTS'13.

Overview

Consider the set of jobs schedulable by the following approaches:

$$FP \overset{[2]}{=} OCBP \overset{[3]}{\subsetneq} MCEDF \subsetneq FPM \qquad STTM$$

[2] Sanjoy K. Baruah, Haohan Li, and Leen Stougie. "Towards the Design of Certifiable Mixed-criticality Systems". In: *Real-Time and Embedded Technology and Applications Symposium*. RTAS'10.

[3] Dario Socci et al. "Mixed Critical Earliest Deadline First". In: *Euromicro Conf. on Real-Time Systems*. ECRTS'13.

Overview

Consider the set of jobs schedulable by the following approaches:

$$FP \overset{[2]}{=} OCBP \overset{[3]}{\subsetneq} MCEDF \subsetneq FPM \subsetneq STTM$$

[2] Sanjoy K. Baruah, Haohan Li, and Leen Stougie. "Towards the Design of Certifiable Mixed-criticality Systems". In: *Real-Time and Embedded Technology and Applications Symposium*. RTAS'10.

[3] Dario Socci et al. "Mixed Critical Earliest Deadline First". In: *Euromicro Conf. on Real-Time Systems*. ECRTS'13.

Overview

Consider the set of jobs schedulable by the following approaches:

$$FP \overset{[2]}{=} OCBP \overset{[3]}{\subsetneq} MCEDF \subsetneq FPM \subsetneq STTM$$

$$Alg: FPM \mapsto STTM$$

[2] Sanjoy K. Baruah, Haohan Li, and Leen Stougie. "Towards the Design of Certifiable Mixed-criticality Systems". In: *Real-Time and Embedded Technology and Applications Symposium*. RTAS'10.

[3] Dario Socci et al. "Mixed Critical Earliest Deadline First". In: *Euromicro Conf. on Real-Time Systems*. ECRTS'13.

The algorithm

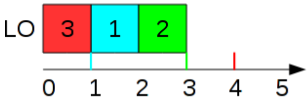
- INPUT:
  FPM scheduling defined by the priority tables PT_{LO} and PT_{HI}
- OUTPUT:
  STTM scheduling defined by two TT table **LO** and **HI***

The algorithm

- INPUT:
  FPM scheduling defined by the priority tables PT_{LO} and PT_{HI}
- OUTPUT:
  STTM scheduling defined by two TT table **LO** and **HI***

- We generate table **LO** by simulating the execution of the LO scenario

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	1	LO	1	1
2	0	3	LO	1	1
3	0	4	HI	1	3



The algorithm

generation of HI*

- Let at time t:
  - $T_j^{LO}(t)$ (resp. $T_j^{HI*}(t)$) = progress of J_j in **LO** (resp. **HI***)
  - $E^{LO}(t)$ = job running in **LO**

The algorithm

generation of HI*

- Let at time t:
  - $T_j^{LO}(t)$ (resp. $T_j^{HI*}(t)$) = progress of J_j in **LO** (resp. **HI***)
  - $E^{LO}(t)$ = job running in **LO**

Generate **HI*** by simulation of HI jobs:

- each job executes for $C(\mathrm{HI})$ time units
- the jobs are scheduled according to PT_{HI}
- a job is enabled by one of the following *rules*:

$$T_j^{LO}(t) = C_j(\mathrm{LO})$$

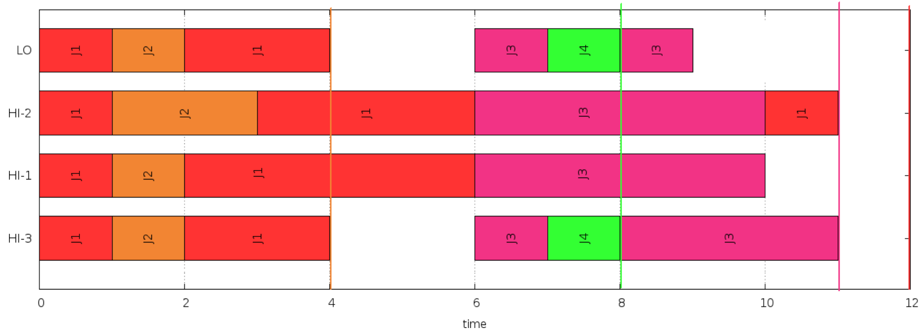
$$T_j^{HI*}(t) < T_j^{LO}(t)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j$$

Example

Job	A	D	χ	$C(LO)$	$C(HI)$
1	0	12	HI	3	5
2	1	4	HI	1	2
3	6	11	HI	2	4
4	7	8	LO	1	1

$$PT_{LO} = J_4 \succ J_2 \succ J_3 \succ J_1$$
$$PT_{HI} = J_2 \succ J_3 \succ J_1$$



Example

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \qquad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \qquad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \qquad (3)$$

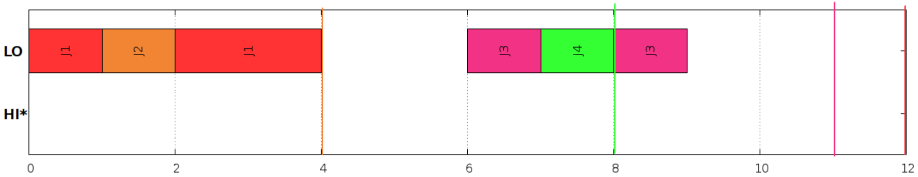
Example

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \quad (1)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$



$t = 0$ $E^{LO}(t) = J_1$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\mathrm{LO})$	$C(\mathrm{HI})$	(1)	(2)	(3)
1	Enabled	0	0	3	5			✓
2								
3								

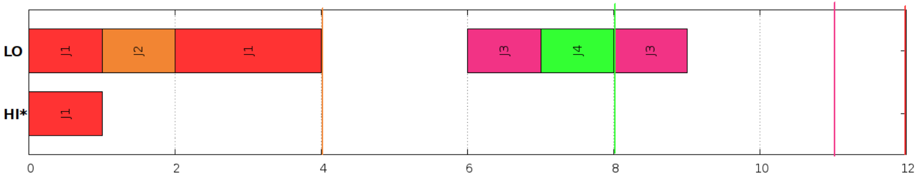
Example

$$T_j^{LO}(t) = C_j(\text{LO}) \qquad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \qquad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \; \wedge E^{LO}(t) = J_j \qquad (3)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$



$t = 1 \qquad E^{LO}(t) = J_2$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\text{LO})$	$C(\text{HI})$	(1)	(2)	(3)
1	Disabled	1	1	3	5			
2	Enabled	0	0	1	2			✓
3								

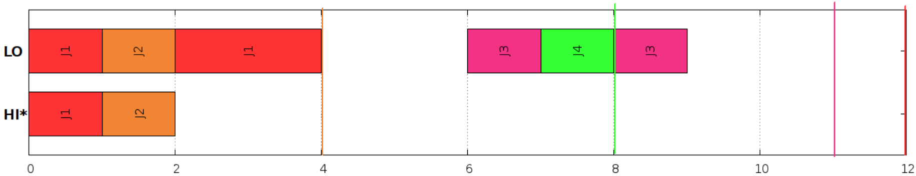
Example

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{LO}(t) = C_j(\text{LO}) \quad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$



$t = 2 \qquad E^{LO}(t) = J_1$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\text{LO})$	$C(\text{HI})$	(1)	(2)	(3)
1	Enabled	1	1	3	5			✓
2	Enabled	1	1	1	2	✓		
3								

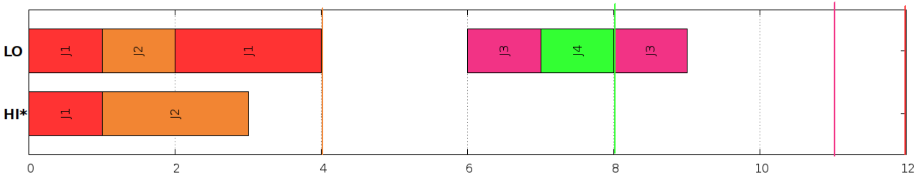
Example

$$T_j^{LO}(t) = C_j(\text{LO}) \quad (1)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$



$t = 3$ $\quad E^{LO}(t) = J_1$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\text{LO})$	$C(\text{HI})$	(1)	(2)	(3)
1	Enabled	1	2	3	5		✓	
2	Term.	2	1	1	2			
3								

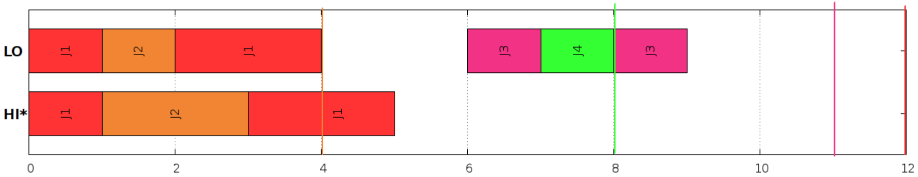
Example

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \quad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$



$t = 5 \qquad E^{LO}(t) = \perp$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\mathrm{LO})$	$C(\mathrm{HI})$	(1)	(2)	(3)
1	Enabled	3	3	3	5	✓		
2	Term.	2	1	1	2			
3								

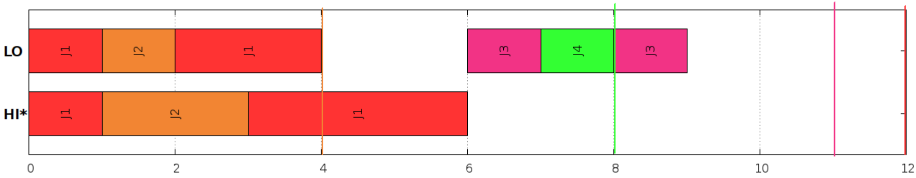
Example

$$T_j^{LO}(t) = C_j(\text{LO}) \quad (1)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$



$t = 6$ $\quad E^{LO}(t) = J_3$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\text{LO})$	$C(\text{HI})$	(1)	(2)	(3)
1	Enabled	4	3	3	5	✓		
2	Term.	2	1	1	2			
3	Enabled	0	0	2	4			✓

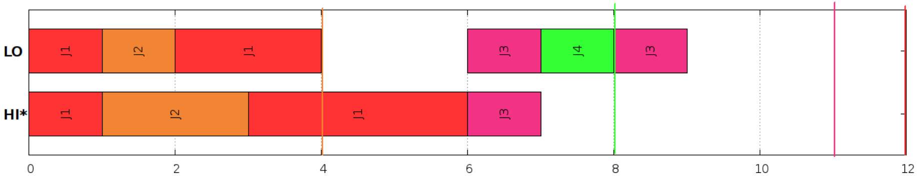
Example

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \qquad (1)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \qquad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \qquad (3)$$



$t = 7 \qquad E^{LO}(t) = J_4$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\mathrm{LO})$	$C(\mathrm{HI})$	(1)	(2)	(3)
1	Enabled	4	3	3	5	✓		
2	Term.	2	1	1	2			
3	Disabled	1	1	2	4			

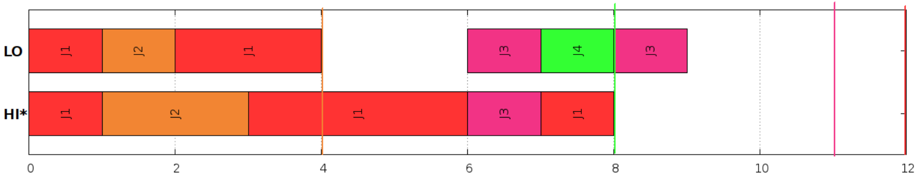
Example

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \qquad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \qquad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \qquad (3)$$



$t = 8 \qquad E^{LO}(t) = J_3$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\mathrm{LO})$	$C(\mathrm{HI})$	(1)	(2)	(3)
1	Term.	5	3	3	5			
2	Term.	2	1	1	2			
3	Enabled	1	1	2	4			✓

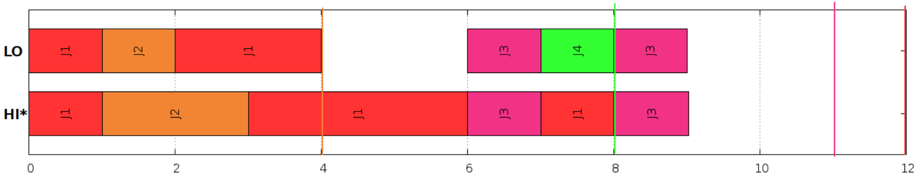
Example

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{LO}(t) = C_j(\mathrm{LO}) \qquad (1)$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \qquad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \qquad (3)$$



$t = 9 \qquad E^{LO}(t) = \perp$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\mathrm{LO})$	$C(\mathrm{HI})$	(1)	(2)	(3)
1	Term.	5	3	3	5			
2	Term.	2	1	1	2			
3	Enabled	2	1	2	4	✓		

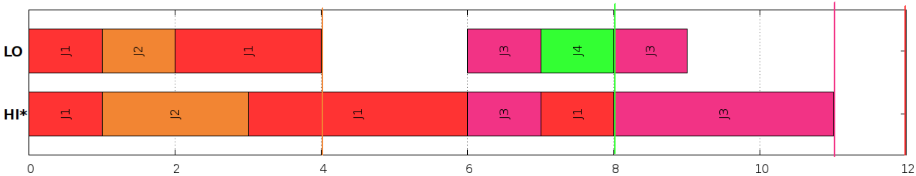
Example

$$T_j^{LO}(t) = C_j(\text{LO}) \quad (1)$$

$$PT_{HI} = J_2 \succ J_3 \succ J_1$$

$$T_j^{HI*}(t) < T_j^{LO}(t) \quad (2)$$

$$T_j^{HI*}(t) = T_j^{LO}(t) \ \wedge E^{LO}(t) = J_j \quad (3)$$



$t = 11 \qquad E^{LO}(t) = \perp$

Job	STATUS	$T^{HI*}(t)$	$T^{LO}(t)$	$C(\text{LO})$	$C(\text{HI})$	(1)	(2)	(3)
1	Term.	5	3	3	5			
2	Term.	2	1	1	2			
3	Term.	4	1	2	4			

Proof of correctness

Theorem

If the FPM policy leads to a feasible schedule, then a switched time triggered schedule that uses **LO** *and* **HI*** *as, respectively, LO-mode and HI-mode table, is a feasible schedule as well.*

Proof of correctness

Theorem

If the FPM policy leads to a feasible schedule, then a switched time triggered schedule that uses **LO** *and* **HI*** *as, respectively, LO-mode and HI-mode table, is a feasible schedule as well.*

the proof is based on the followings:

Lemma

If at any time we switch from **LO** *to* **HI***, *then all the unterminated jobs will have enough time reserved in* **HI*** *to terminate their work.*

Proof of correctness

Theorem

If the FPM policy leads to a feasible schedule, then a switched time triggered schedule that uses **LO** *and* **HI*** *as, respectively, LO-mode and HI-mode table, is a feasible schedule as well.*

the proof is based on the followings:

Lemma

If at any time we switch from **LO** *to* **HI***, *then all the unterminated jobs will have enough time reserved in* **HI*** *to terminate their work.*

Theorem

For all HI jobs J there exists a basic scenario of the FPM scheduling where J terminates no earlier than in **HI***

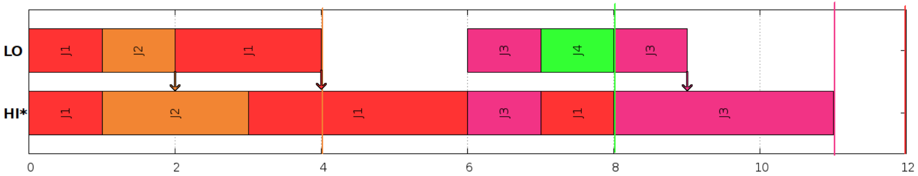
correctness of example

It is easy to show that the lemma is true for our example

Lemma

*If at any time we switch from **LO** to **HI***, then all the unterminated jobs will have enough time reserved in **HI*** to terminate their work.*

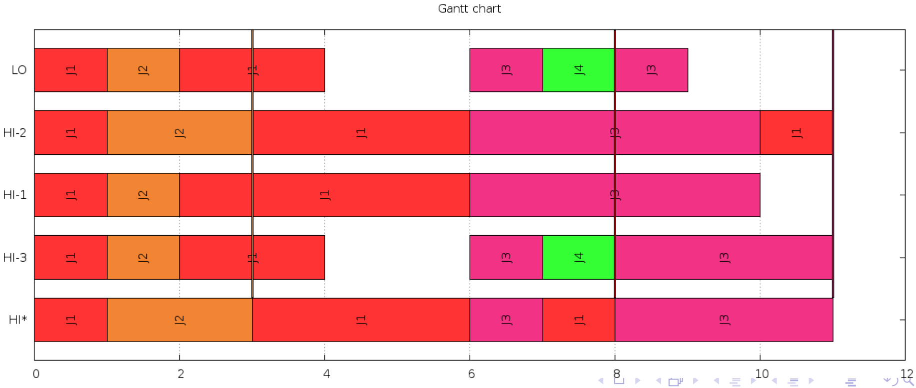
$C_1(\mathrm{HI}) = 5$ $C_2(\mathrm{HI}) = 2$ $C_3(\mathrm{HI}) = 4$



correctness of example

Theorem

For all HI jobs J there exists a basic scenario of the FPM scheduling where J terminates no earlier than in **HI***



- Conclusions
  - Proof that $FPM \subset STTM$
  - Algorithm to translate FPM solution into $STTM$
    - this can make certification easier

- Conclusions
  - Proof that $FPM \subset STTM$
  - Algorithm to translate FPM solution into $STTM$
    - this can make certification easier

- Future work
  - More than two criticality levels
  - Multiprocessor