

First Labs on Real-Time Scheduling

Hamza Mouddene

November 26, 2021

Exercise 1

- Let's assume the following task configuration on one core using *deadline monotonic*.

| | First release | WCET | D | P |
|-------|---------------|------|---|----|
| T_1 | 0 | 2 | 5 | 5 |
| T_2 | 0 | 2 | 4 | 10 |

- The task set is schedulable if you computed the scheduling on the feasibility interval.
- For the task T_1 is 4 time units and for T_2 is 2 time units.

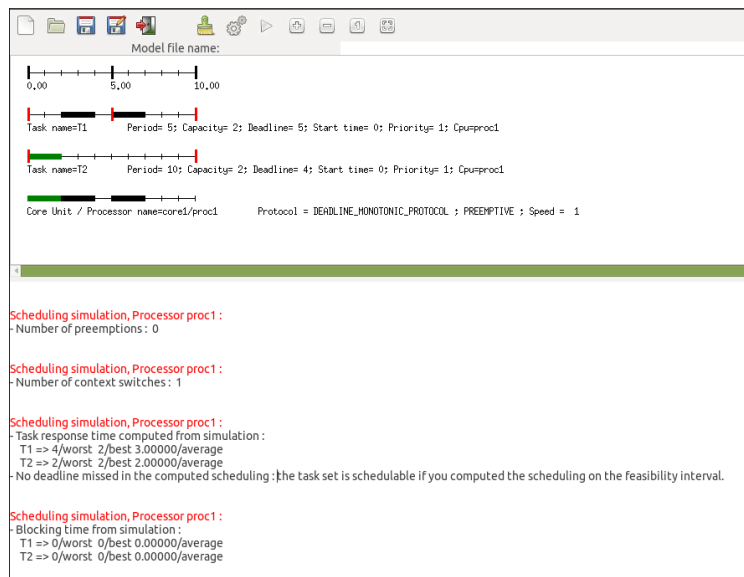


Figure 1: Simulation of the task configuration

- Now, we will fix a priority order in the task configuration, we will giving the T_1 the highest priority, so the scheduling simulation gives the following result:

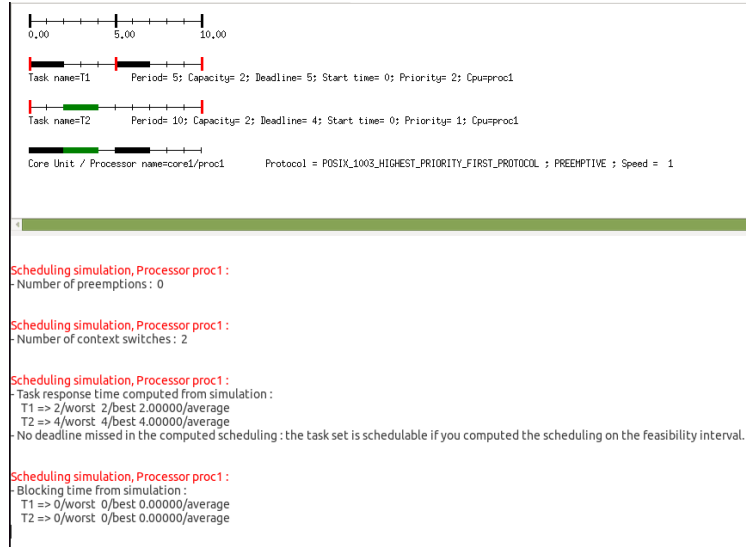


Figure 2: Simulation of the task configuration with a fixed priority

The task set is schedulable if you computed the scheduling on the feasibility interval.

Exercise 2

- Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|----|----|
| T_1 | 0 | 2 | 6 | 6 |
| T_2 | 0 | 4 | 10 | 10 |
| T_3 | 0 | 2 | 20 | 20 |

It is schedulable by a fixed priority.

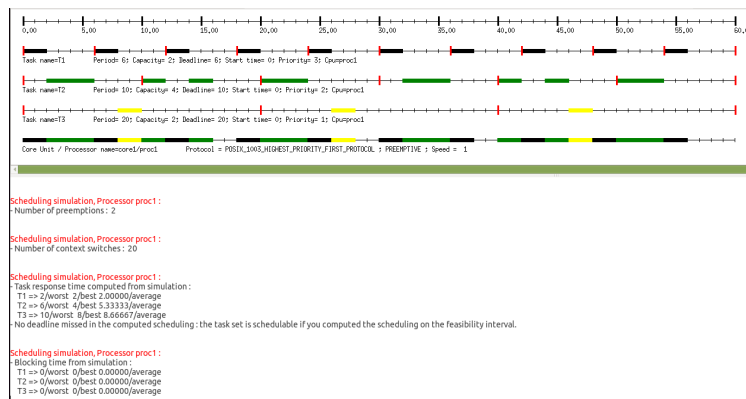


Figure 3: Simulation of the task configuration - Fixed priority algorithm

2. Now, Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|----|----|
| T_1 | 0 | 2 | 6 | 6 |
| T_2 | 0 | 4 | 10 | 10 |
| T_3 | 0 | 2 | 9 | 20 |

It is not schedulable by a fixed priority.

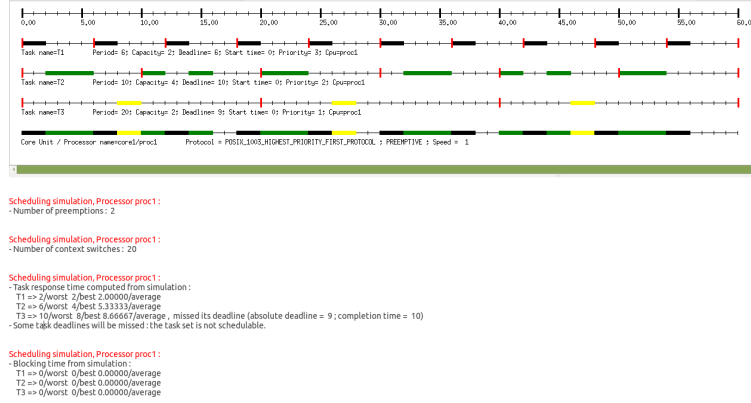


Figure 4: Simulation of the task configuration - Fixed priority algorithm

It is schedulable by a Earliest Deadline First.

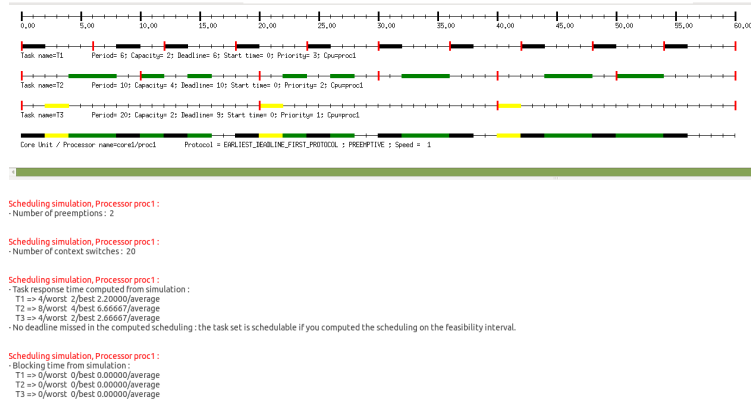


Figure 5: Simulation of the task configuration - Earliest Deadline First

3. Now, Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|---|----|
| T_1 | 0 | 2 | 6 | 6 |
| T_2 | 0 | 4 | 8 | 10 |
| T_3 | 0 | 2 | 9 | 20 |

It is not schedulable by a fixed priority.

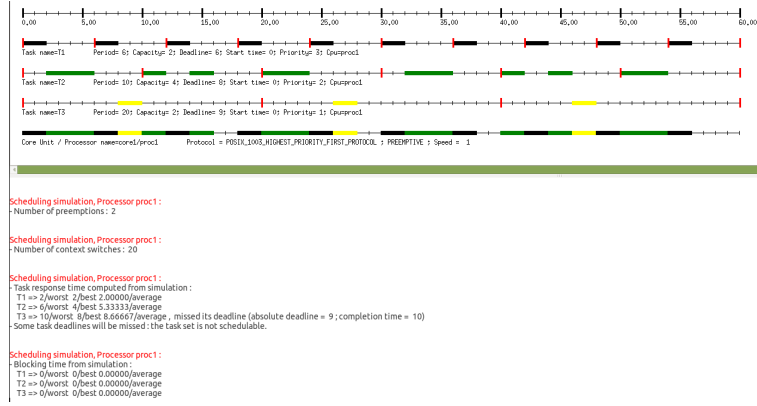


Figure 6: Simulation of the task configuration - Fixed priority algorithm

It is schedulable by a Earliest Deadline First.

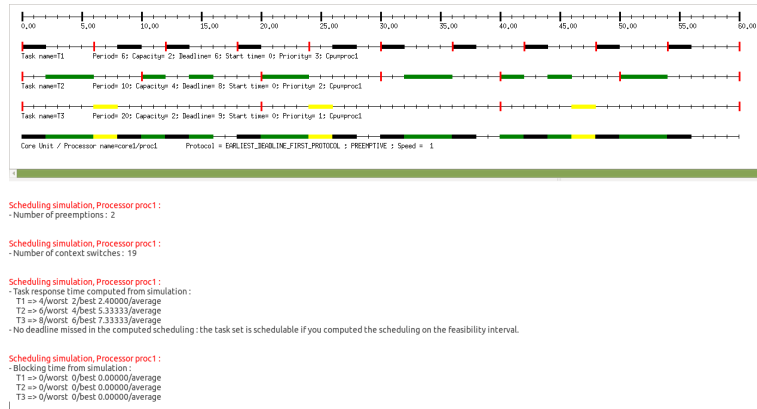


Figure 7: Simulation of the task configuration - Earliest Deadline First

Exercise 3

Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|---|---|
| T_1 | 0 | 3 | 8 | 8 |
| T_2 | 0 | 4 | 9 | 9 |

As the simulation shows, the Least Laxity First algorithm needs more preemption's compared to Earliest Deadline First algorithm, that implies that the Least Laxity First algorithm consumes more resources.

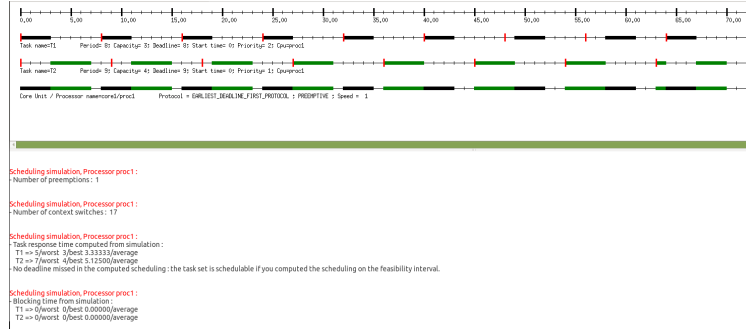


Figure 8: Simulation of the task configuration - Earliest Deadline First

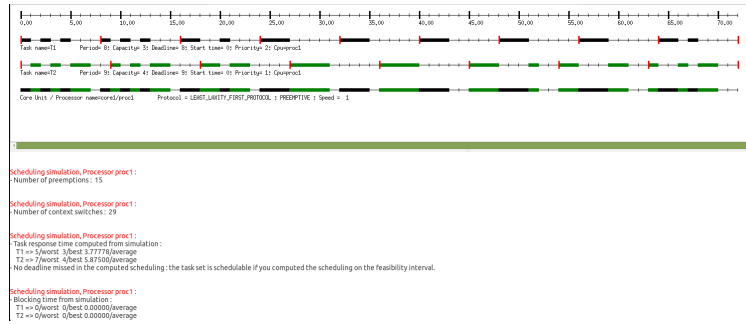


Figure 9: Simulation of the task configuration - Least Laxity First

Exercise 4

- Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|---|---|
| T_1 | 0 | 1 | 3 | 3 |
| T_2 | 0 | 3 | 9 | 9 |

Here, we can see that the response time of T_1 is shorter in non preemptive mode, in contrary the the response time of T_1 is longer in non preemptive mode. The task set is schedulable if you computed the scheduling on the feasibility interval. But the only difference between the both modes is the task response time as we see in the following figures.

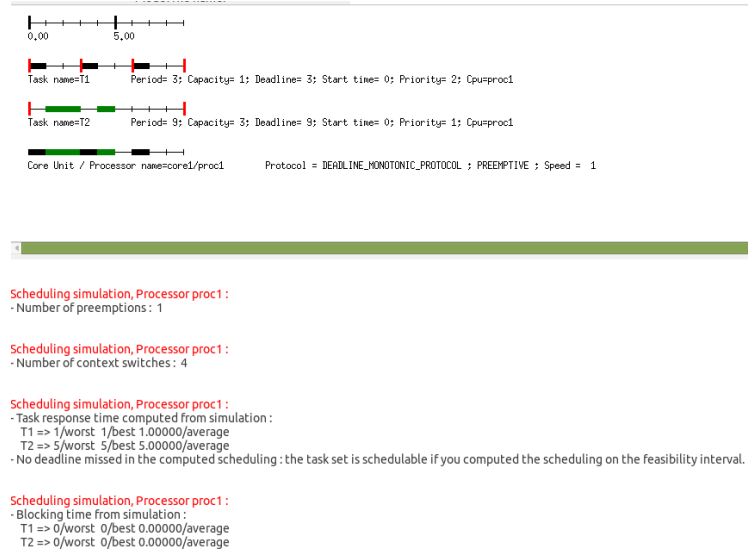


Figure 10: Simulation of the task configuration - Preemptive Rate Monotonic

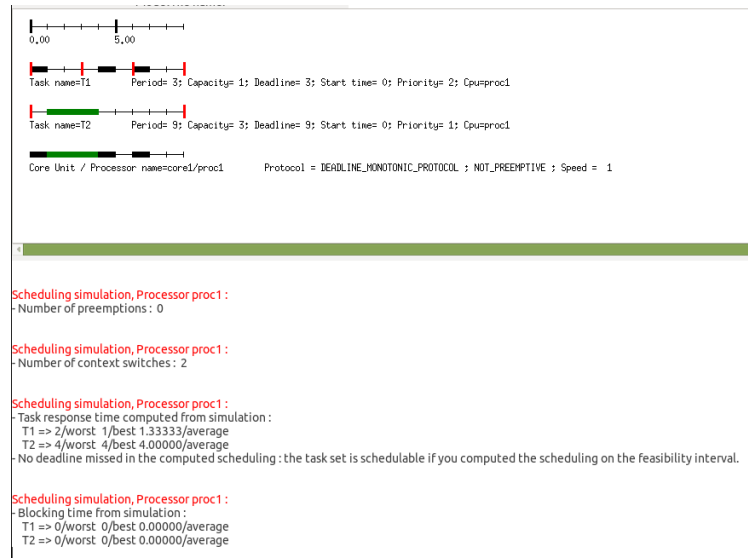


Figure 11: Simulation of the task configuration - Non Preemptive Rate Monotonic

2. Let's assume the following independent task configuration.

| | First release | WCET | D | P |
|-------|---------------|------|---|---|
| T_1 | 0 | 2 | 3 | 3 |
| T_2 | 0 | 3 | 9 | 9 |

In this example, the task set is schedulable only for preemptive mode, in not preemptive mode T_2 fails to achieve its deadline that make the task set not schedulable.

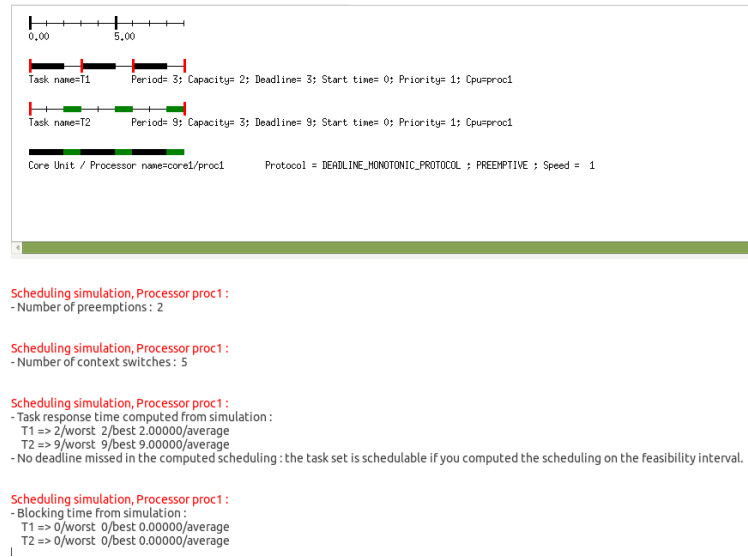


Figure 12: Simulation of the task configuration - Preemptive Rate Monotonic

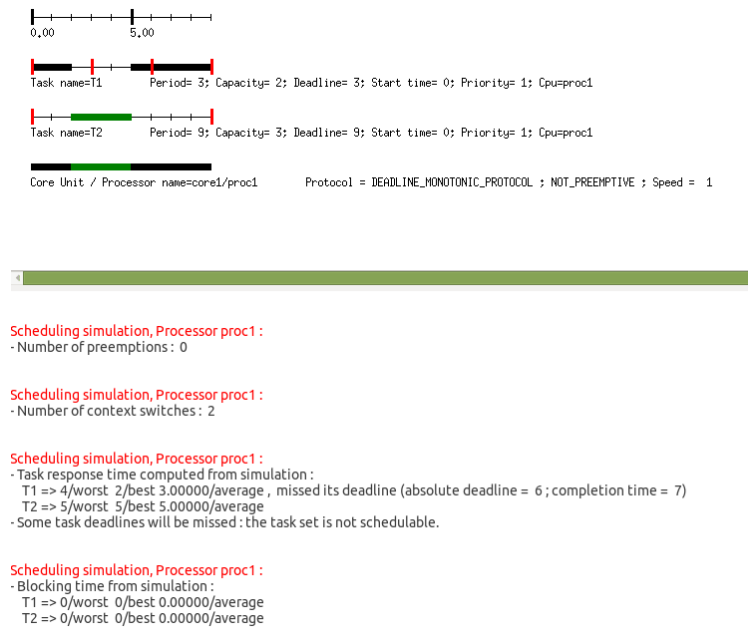


Figure 13: Simulation of the task configuration - Non Preemptive Rate Monotonic

Exercise 5

Let's assume the following dependant task configuration:

We have the following precedence constraints:

- T_1 and T_2 have to complete execution before T_3 starts,
- T_3 has to complete execution before T_4 and T_5 .

| | First release | WCET | D | P |
|-------|---------------|------|----|----|
| T_1 | 0 | 1 | 10 | 10 |
| T_2 | 2 | 1 | 8 | 10 |
| T_3 | 0 | 2 | 10 | 10 |
| T_4 | 0 | 1 | 10 | 10 |
| T_5 | 0 | 1 | 10 | 10 |
| T_6 | 0 | 8 | 15 | 20 |

1. It is not schedulable using the rate monotonic approach presented in the lectures, because the task must have period equal to deadline.

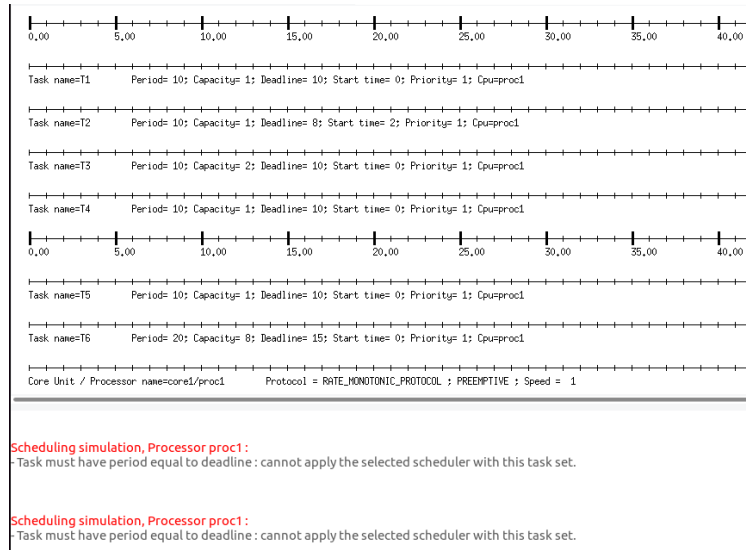


Figure 14: Simulation of the task configuration - Rate Monotonic Protocol

2. It is schedulable using the Earliest Deadline First approach presented in the lectures, as we can see on the following result:

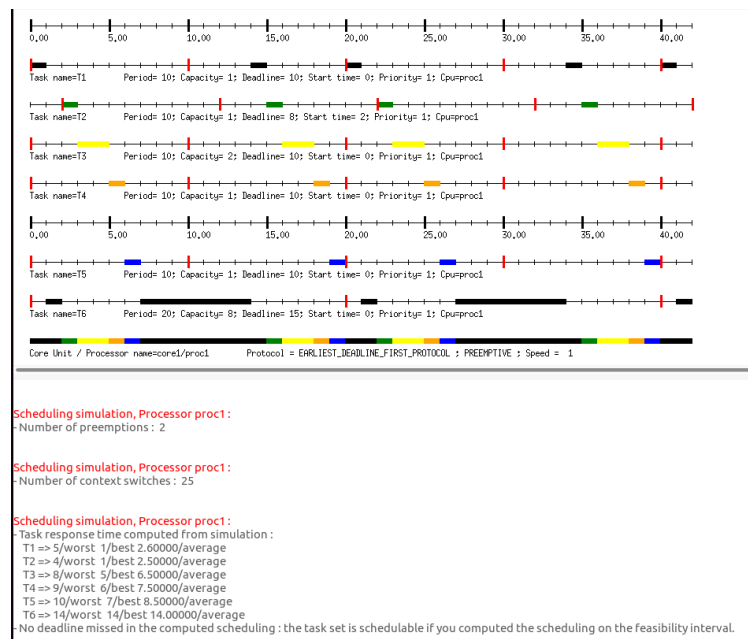


Figure 15: Simulation of the task configuration - Earliest Deadline First Protocol