

# SimSo: A Simulation Tool to Evaluate Real-Time Multiprocessor Scheduling Algorithms

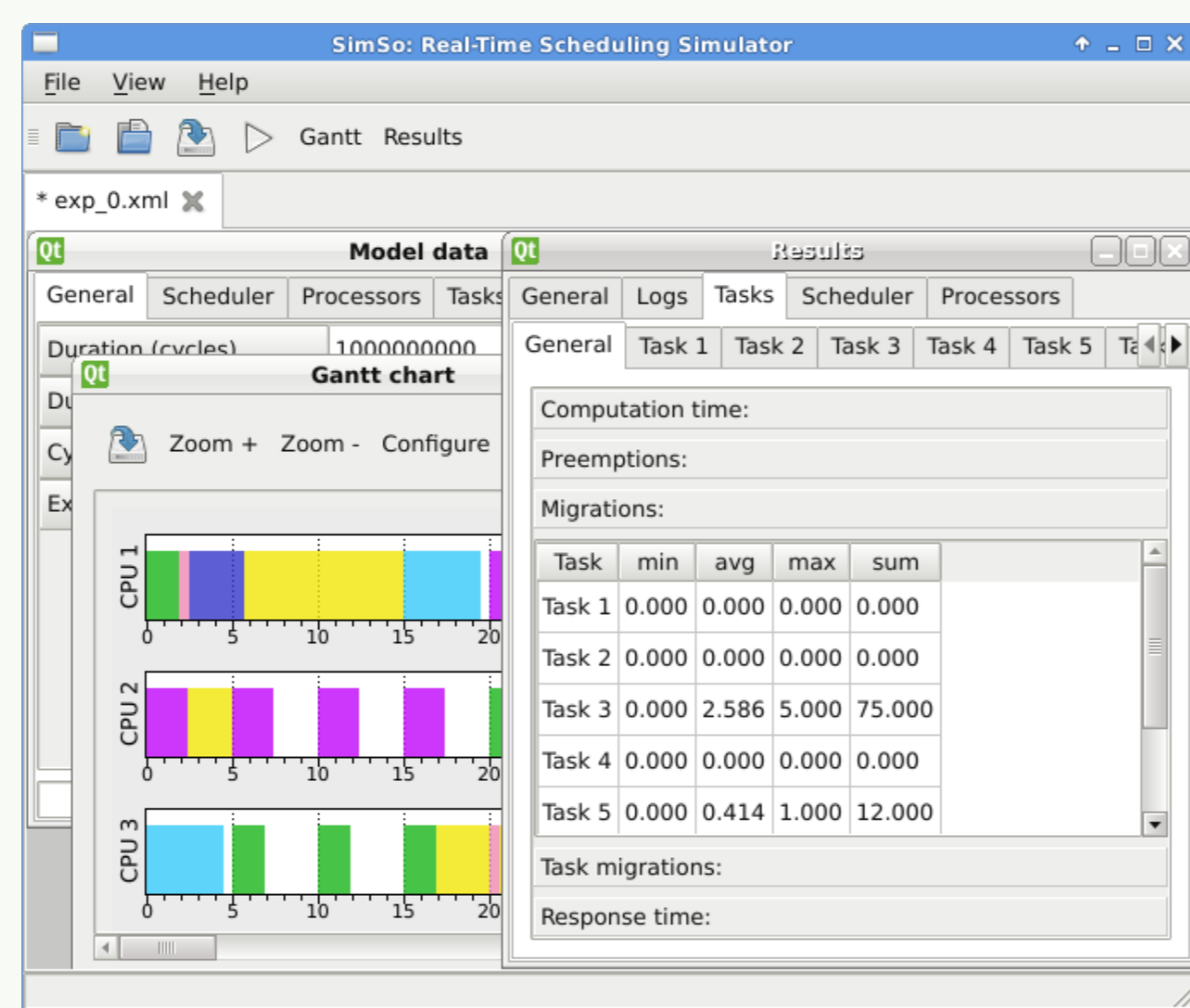
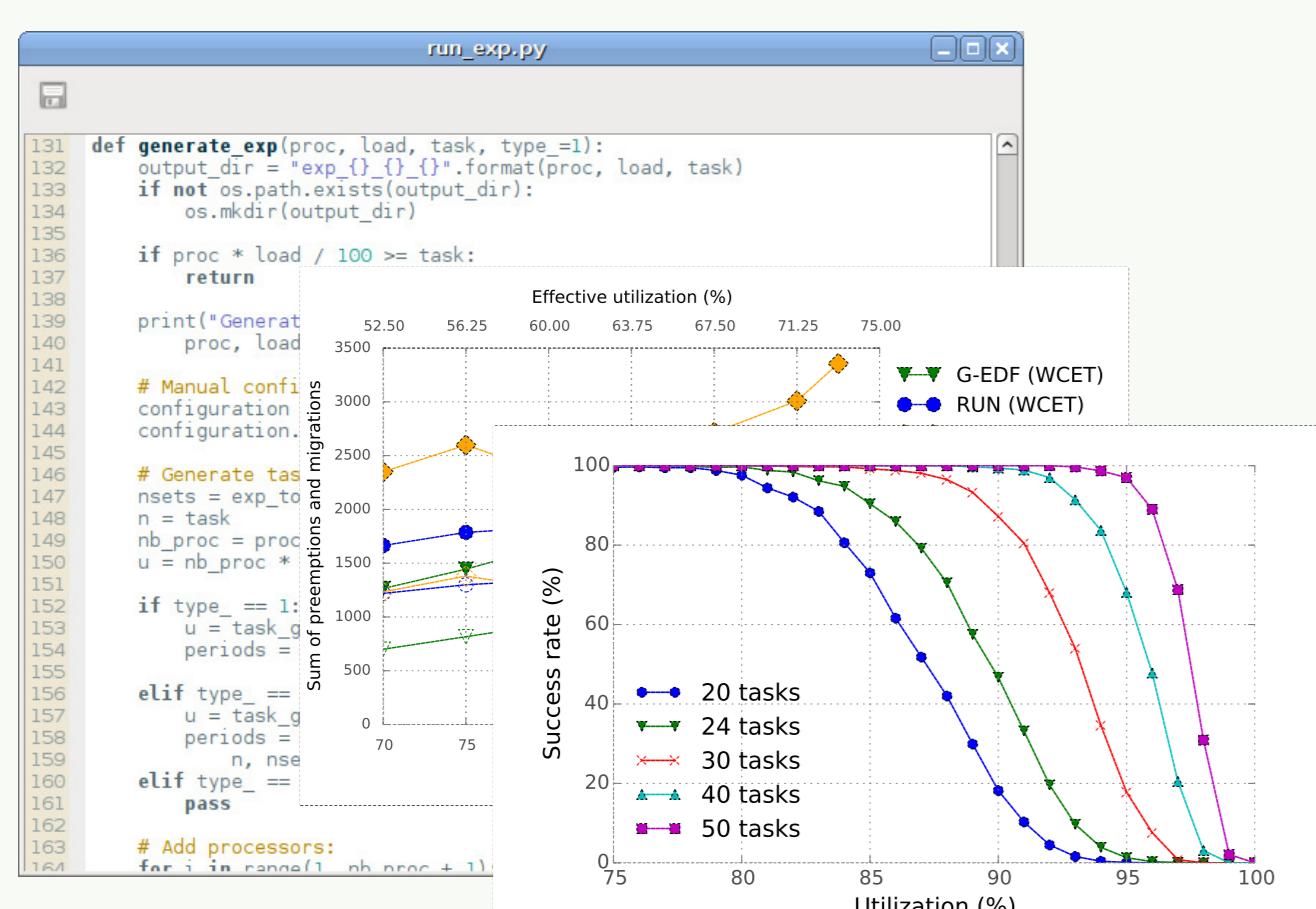
Maxime Chéramy<sup>1</sup>, Pierre-Emmanuel Hladik<sup>1</sup> and Anne-Marie Déplanche<sup>2</sup>

<sup>1</sup>CNRS, LAAS, Univ de Toulouse, INSA, Toulouse, France

<sup>2</sup>Université de Nantes, IRCCyN UMR CNRS 6597, ECN, Nantes, France

## Easy to Use

With a Graphical User Interface:



... or as a Python Module in order to automate multiple simulations.

## 25+ Available Schedulers

### Uniprocessor:

RM, DM, FP, EDF, LLF, M-LLF, Static-EDF, CC-EDF

### Partitioned:

Any uniprocessor scheduler combined with various bin packing algorithms

### Global:

G-RM, G-EDF, G-FL, EDF-US, PriD, EDZL, LLF, M-LLF, U-EDF

### PFair:

PD<sup>2</sup>, ER-PD<sup>2</sup>

### DP-Fair and BFair:

LLREF, LRE-TL, DP-WRAP, BF, NVNLF

### Semi-Partitioned:

EDHS, EKG, RUN

## SimSo

SimSo is a simulator that is designed to facilitate the evaluation of scheduling algorithms.

Open Source: <http://homepages.laas.fr/mcheramy/simso/>

## Conducting an Evaluation Campaign

### Writing a Scheduler

A scheduler for SimSo is a Python class that inherits from an abstract class.

This scheduler will be loaded dynamically by SimSo.

### Execution Time Models

The computation time of the jobs are decided by a model during simulation.

Current models are: WCET, ACET, fixed preemption delays, cache related delays.

### Generation of a Taskset

Various generators are available:

#### Task utilizations:

RandFixedSum, UUniFast-Discard and other generators.

#### Task periods:

Uniform, Log-Uniform, Discrete.

A taskset can also be generated using an external tool.

### Simulation

SimSo can be used from a script to automate the simulations and extract data.

It can run thousands of simulations per hour.

### Analysis

The output of a simulation is a trace of all the scheduling events.

Some functions ease the retrieval of common metrics (preemptions, migrations, response times, exceeded deadlines, etc).