Simulation of Real-Time Scheduling with Various Execution Time Models

Maxime Chéramy, Pierre-Emmanuel Hladik, Anne-Marie Déplanche and Sébastien Dubé

maxime.cheramy@laas.fr

June 19, 2014

9th IEEE International Symposium on Industrial Embedded Systems (SIES)
Many Real-Time Multiprocessor Scheduling Algorithms

Requirement: evaluate their behavior and performance.

Objective: propose a friendly and dedicated simulation-based tool.

Simulation of Real-Time Scheduling
Requirement: evaluate their behavior and performance.
Requirement: evaluate their behavior and performance.

Objective: propose a friendly and dedicated simulation-based tool.
SimSo: a Multiprocessor Real-Time Scheduling Simulator
SimSo: a Multiprocessor Real-Time Scheduling Simulator

Open Source
Easy to use

SimSo
SimSo: a Multiprocessor Real-Time Scheduling Simulator

Open Source
Easy to use

25+ schedulers

SimSo

Simulation of Real-Time Scheduling
SimSo: a Multiprocessor Real-Time Scheduling Simulator

- Open Source
- Easy to use
- 25+ schedulers
- Various Execution Time Models
SimSo: a Multiprocessor Real-Time Scheduling Simulator

- Open Source
- Easy to use
- 25+ schedulers
- Various Execution Time Models
- Graphical User Interface

SimSo

Utilization (%)
0
500
1000
1500
2000
2500
3000
3500

Sum of preemptions and migrations

G-EDF (WCET)
RUN (WCET)
U-EDF (WCET)

Effective utilization (%)

G-EDF (ACET)
RUN (ACET)
U-EDF (ACET)

Success rate (%)

20 tasks
24 tasks
30 tasks
40 tasks
50 tasks
SimSo: a Multiprocessor Real-Time Scheduling Simulator

- 25+ schedulers
- Various Execution Time Models
- Open Source
- Easy to use
- 25+ schedulers
- Various Execution Time Models

Graphical User Interface

Script Mode

Utilization (%)

Sum of preemptions and migrations

G-EDF (WCET)
RUN (WCET)
U-EDF (WCET)

Effective utilization (%)

G-EDF (ACET)
RUN (ACET)
U-EDF (ACET)

Success rate (%)

20 tasks
24 tasks
30 tasks
40 tasks
50 tasks

Simulation of Real-Time Scheduling
Execution Time Model 1: Using the WCET vs the ACET

Using the WCET is pessimistic

Upper-bound rarely reached in practice

Unlikely that all the jobs meet their WCET at the same time
Execution Time Model 1: Using the WCET vs the ACET

Using the WCET is pessimistic

- Upper-bound rarely reached in practice
- Unlikely that all the jobs meet their WCET at the same time

It makes sense for schedulability analyses.
Using the WCET is pessimistic

- Upper-bound rarely reached in practice
- Unlikely that all the jobs meet their WCET at the same time

It makes sense for schedulability analyses.

But we discuss its use for empirical studies focusing on other aspects, including the number of preemptions and migrations.
We ran several simulations using the WCET and a random duration lower than the WCET.
We ran several simulations using the WCET and a random duration lower than the WCET.

Using the WCET could introduce a bias in the evaluation:

- Gives an advantage to schedulers that use the WCET
- Some schedulers could benefit from shorter execution times
- Some schedulers are robust to punctual overloads
Caches may have a significant impact on the computation time of the jobs, which depends on the scheduling decisions.
Caches may have a significant impact on the computation time of the jobs, which depends on the scheduling decisions.

There are cache-aware schedulers that require to take into consideration these cache effects.
Caches may have a significant impact on the computation time of the jobs, which depends on the scheduling decisions.

There are cache-aware schedulers that require to take into consideration these cache effects.

⇒ As a consequence, we want to simulate the cache effects.
Extension of the Liu&Layland task model with:

- SDP: Stack Distance Profile
- MIX / API: ratio of memory access per instruction
- CPI: Cycles per Instruction
- Number of executed instructions

Extension of the Liu&Layland task model with:

- SDP: Stack Distance Profile\(^a\)
- MIX / API: ratio of memory access per instruction
- CPI: Cycles per Instruction
- Number of executed instructions


Integration of statistical cache models\(^a\) to determine the duration of the jobs dynamically, during the simulation.

\(^a\)D. Chandra et al. “Predicting inter-thread cache contention on a chip multi-processor architecture”. In: *Proc. of HPCA*. 2005.
Simulation of Real-Time Scheduling with Various Execution Time Models

Maxime Chéramy, Pierre-Emmanuel Hladik, Anne-Marie Déplanche and Sébastien Dubé

CNRS, LAAS, Tete de Tarbes, TSA, Toulouse, France
Université de Toulouse, INSA, Toulouse, France
Data Engineering France S.A.S., Toulouse, France

Abstract

SimSo is a simulator that aims at facilitating the experimental evaluation of real-time scheduling algorithms. One of its features is to present various task execution time models. Interestingly, it leads us to observe that using the WCET in empirical simulations could introduce a bias in the results. It also enables to take into consideration the impact of the caches on the computation time of the tasks in order to integrate these effects into scheduling analyses.

Cache Effects

Cache related preemption delays are often greater than the OS overheads [2]. On a system with shared caches, a job may slow down another one running on a different processor.

Several schedulers take into consideration the caches (cache-aware schedulers), either by encouraging/discouraging some non-preemptible, or by using cache space isolation techniques.

Figure 1: Graphical User Interface of SimSo

• Real-Time Scheduling Simulator
• Multiprocessor extension
• Easy to use and to extend
• Open Source (and written in Python)
• More than 25 scheduling algorithms

Figure 2: Simulation using WCET and ACET.

• Execution Time Models
• Cache Effects

The computation time of the jobs is defined by SimSo entities called “Execution Time Models.” SimSo provides the following ones:

- WCET: the computation time is the given worst-case execution time.
- ACET: the computation time is randomly chosen using a normal distribution and bounded by the WCET.
- Fixed Penalty: the computation time is extended when a preemption occurs.
- Cache Model: the computation time is defined by statistical cache models and takes into account the memory access behavior of the tasks.

Discussion about the WCET

Simulating a system so that all tasks meet their worst-case execution time at each job is very pessimistic, since the worst-case is an upper-bound that is rarely reached in practice and it is even more unlikely that it is reached by all the active jobs at the same time.

Simulating Cache Effects

SimSo accepts as input an extended form of its 4Ms and Leachland task model [1], which could not be valued; additional informations to characterize the memory behavior of the tasks:

• Stack Distance Profile: The average number of instructions needed to execute an instruction.
• MIX or API: The number of memory references per instruction.
• CPE: The average number of cycles needed to execute an instruction.
• Number of instructions: The average number of instructions executed.

The cache memory organization and characteristics of the hardware architecture being specified, SimSo is then able to emulate the cache effects on task execution using statistical models [3]. The accuracy of the results is yet to be confirmed.

Bibliography


Thank you for your attention.

Any question? → Come to the poster session, I’d be pleased to discuss this further with you.