

Speed And Accuracy Dilemma In NoC Simulation: What About Memory Impact?

Manuel Selva Abdoulaye Gamatié David Novo Gilles
Sassatelli

LIRMM (CNRS and University of Montpellier)

18 January 2016



Context

Manycore processors integrating a NoC are there

- ▶ Intel Xeon Phi
- ▶ Kalray MPPA2-256
- ▶ TILE-Gx72

NoC simulation tools are needed (and already there)

- ▶ **Booksim**, NoCTweak, Garnet, Noxim, **McSim**
- ▶ The **perfect** simulator is both fast and accurate
- ▶ Speed/accuracy dilemma

Context

Manycore processors integrating a NoC are there

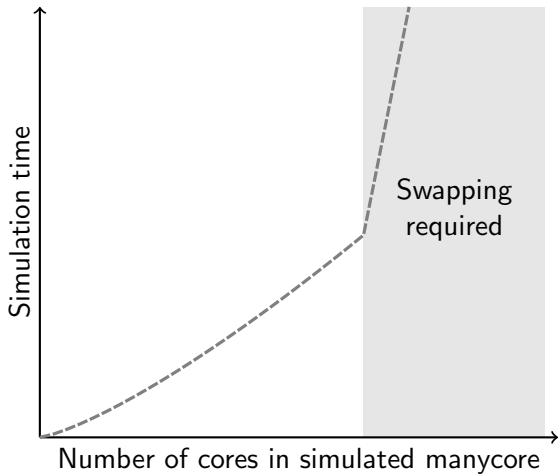
- ▶ Intel Xeon Phi
- ▶ Kalray MPPA2-256
- ▶ TILE-Gx72

NoC simulation tools are needed (and already there)

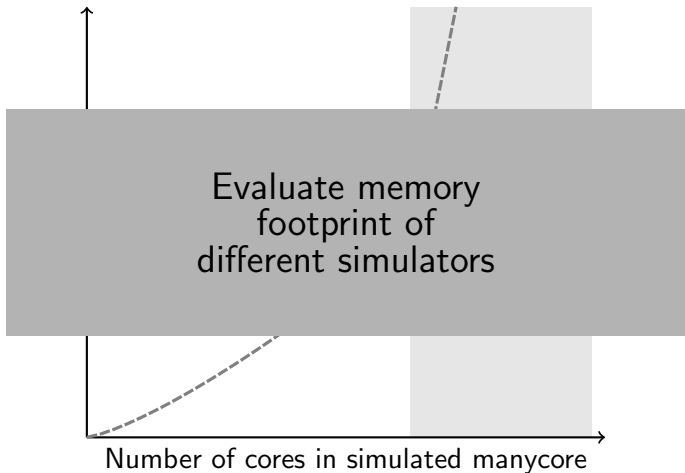
- ▶ **Booksim**, NoCTweak, Garnet, Noxim, **McSim**
- ▶ The **perfect** simulator is both fast and accurate
- ▶ Speed/accuracy dilemma

What about memory footprint?

Why Care About Memory Footprint?



Why Care About Memory Footprint?



Outline

Considered Simulators

Impact Of Accuracy On Memory Footprint

Impact Of Programming Abstraction On Memory Footprint

Conclusions and Perspectives

Considered Simulators - 2 Criteria

Accuracy

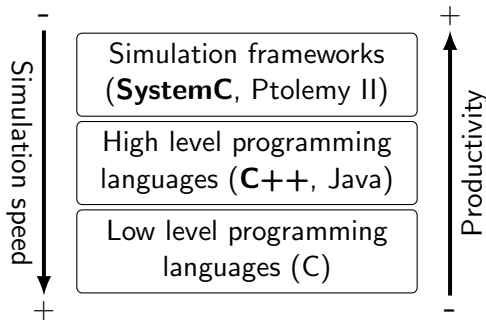
- ▶ Bit-accurate
- ▶ **Cycle-accurate**
- ▶ **Transactional Level Modeling (TLM)**

Considered Simulators - 2 Criteria

Accuracy

- ▶ Bit-accurate
- ▶ **Cycle-accurate**
- ▶ **Transactional Level Modeling (TLM)**

Programming abstraction

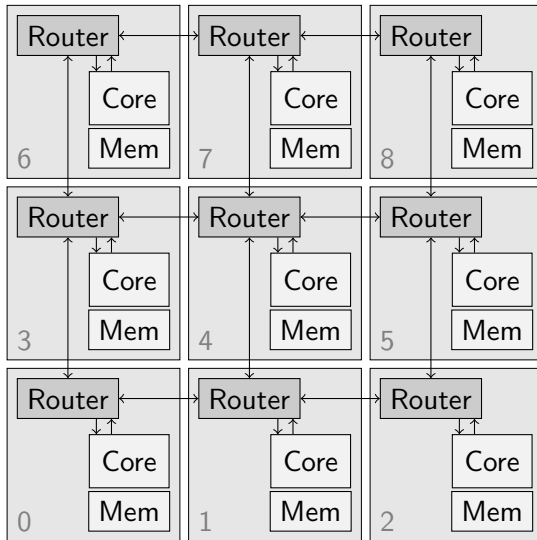


Considered Simulators

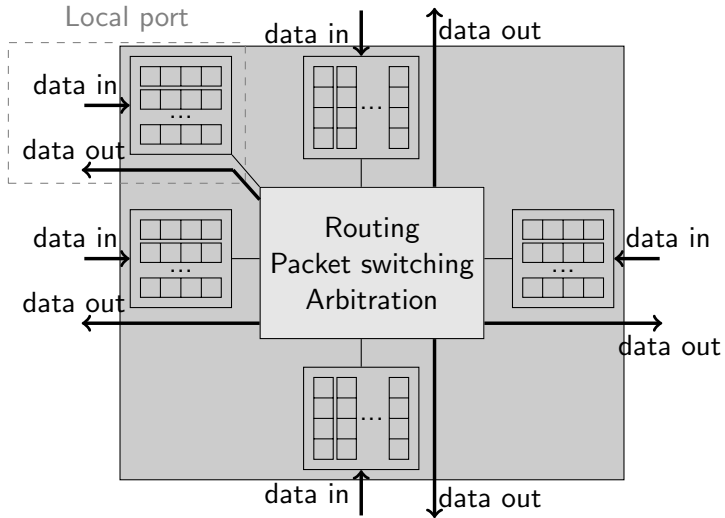
<i>Simulator</i>	<i>Accuracy</i>	<i>Programming abstraction</i>	<i>Injector</i>
<i>McSim-TLM</i>	TLM	SystemC	Application Model
<i>McSim-CA</i>	Cycle-accurate	SystemC	Application Model
<i>Booksim</i>	Cycle-accurate	C++	Random uniform

McSim-CA is based on NoCTweak

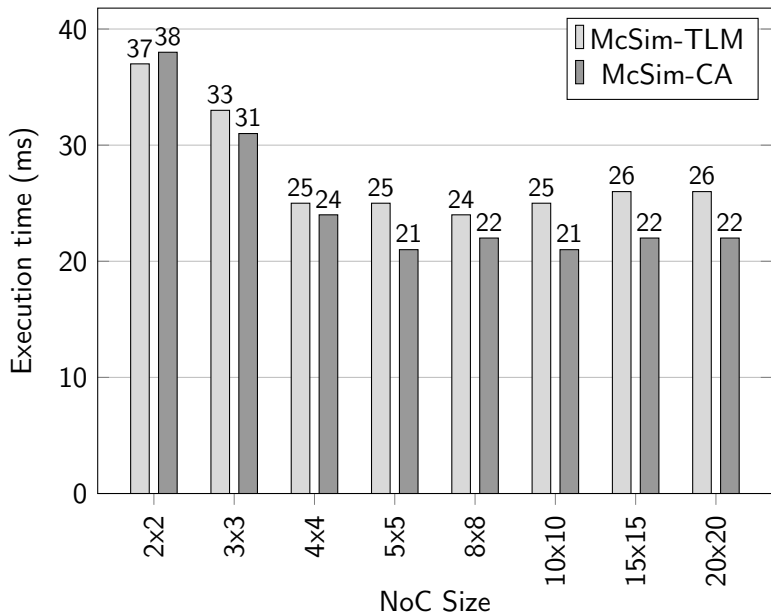
Simulated Hardware - Distributed Memory System



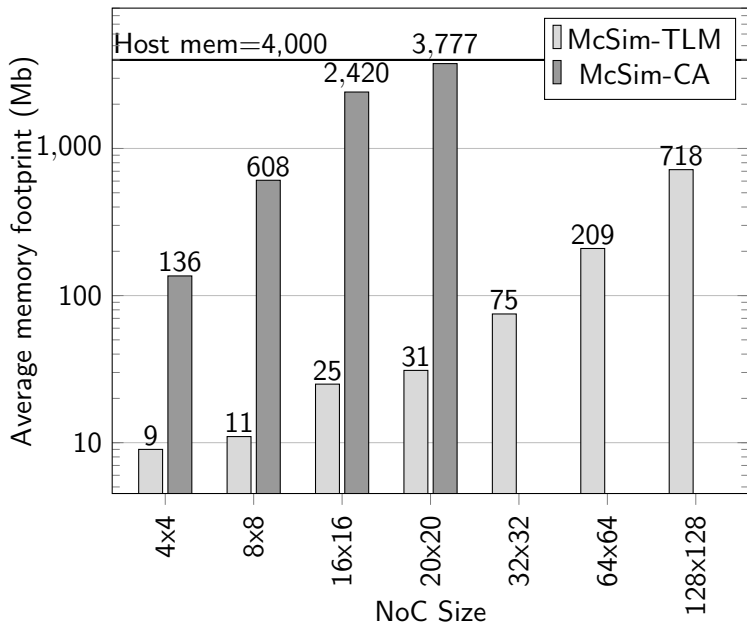
Simulated Hardware - Priority Based Routers



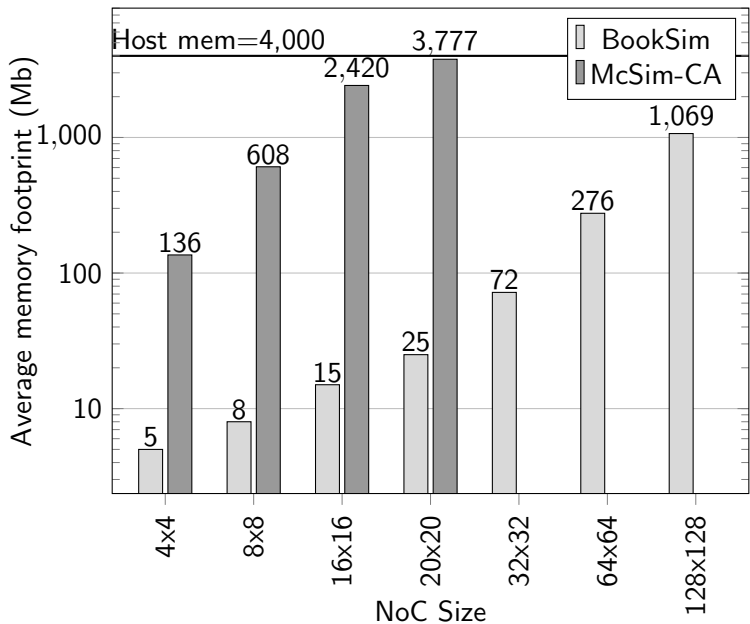
McSim-TLM vs McSim-CA - Accuracy



McSim-TLM vs McSim-CA - Memory Footprint



McSim-CA vs Booksim - Memory Footprint



Deep Memory Footprint Analysis

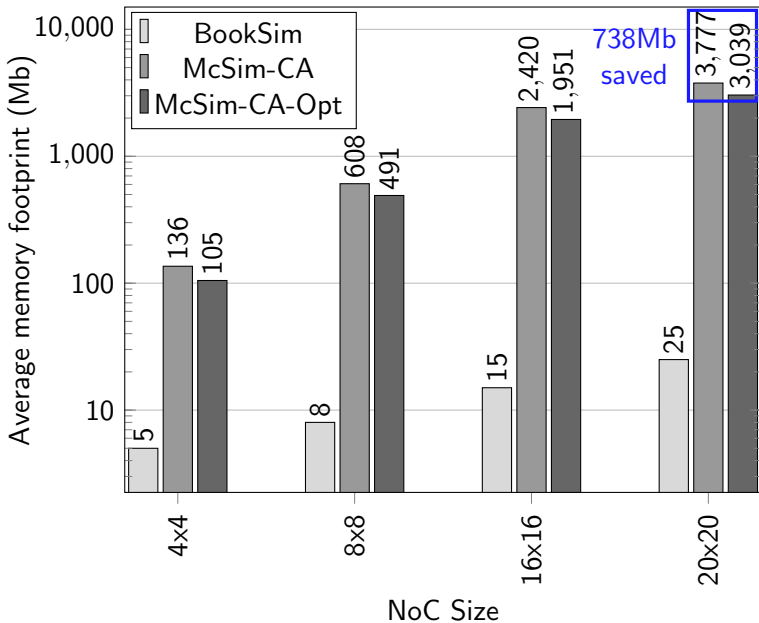
A lot of objects

- ▶ Few big objects accounting for 1% of footprint
- ▶ A lot of small SystemC objects (3,500,000 for 20x20)

Accellera implementation

- ▶ Each SystemC object has a unique name
- ▶ Debug purposes
- ▶ Required by the standard

Optimized Accellera - Memory Footprint



Conclusion

From TLM to cycle-accurate

- ▶ **Costs memory** in addition to CPU

Cycle-accurate concerns

- ▶ Programming abstraction **costs memory** in addition to CPU
- ▶ SystemC object names can consume **a lot** of memory

Perspectives

- ▶ Evaluate memory footprint of other simulators
- ▶ Perform lazy allocation in SystemC?

References I

- ▶ N. Agarwal, T. Krishna, L. S. Peh, and N. K. Jha.
Garnet: A detailed on-chip network model inside a full-system simulator.
In Performance Analysis of Systems and Software, 2009. ISPASS 2009. IEEE International Symposium on, pages 33–42, April 2009.
- ▶ V. Catania, A. Mineo, S. Monteleone, M. Palesi, and D. Patti.
Noxim: An open, extensible and cycle-accurate network on chip simulator.
In Application-specific Systems, Architectures and Processors (ASAP), 2015 IEEE 26th International Conference on, pages 162–163, July 2015.
- ▶ L. S. Indrusiak and O. M. dos Santos.
Fast and accurate transaction-level model of a wormhole network-on-chip with priority preemptive virtual channel arbitration.
In Design, Automation Test in Europe Conference Exhibition (DATE), 2011, pages 1–6, March 2011.
- ▶ Leandro Soares Indrusiak, James Harbin, and Osmar Marchi Dos Santos.
Fast simulation of networks-on-chip with priority-preemptive arbitration.
ACM Trans. Des. Autom. Electron. Syst., 20(4):56:1–56:22, September 2015.

References II

- ▶ Nan Jiang, D.U. Becker, G. Michelogiannakis, J. Balfour, B. Towles, D.E. Shaw, J. Kim, and W.J. Dally.
A detailed and flexible cycle-accurate network-on-chip simulator.
In Performance Analysis of Systems and Software (ISPASS), 2013 IEEE International Symposium on, pages 86–96, April 2013.
- ▶ Khalid Latif, Manuel Selva, Charles Effiong, Roman Ursu, Abdoulaye Gamatie, Gilles Sassatelli, Leonardo Zordan, Luciano Ost, Piotr Dziurzanski, and Leandro Soares Indrusiak.
Design space exploration for complex automotive applications: An engine control system case study.
In Proceedings of the 2016 Workshop on Rapid Simulation and Performance Evaluation: Methods and Tools, RAPIDO '16, pages 2:1–2:7, New York, NY, USA, 2016. ACM.
- ▶ L. Lehtonen, E. Salminen, and T. D. Hmlinen.
Analysis of modeling styles on network-on-chip simulation.
In NORCHIP, 2010, pages 1–4, Nov 2010.
- ▶ Gunar Schirner and Rainer Dömer.
Quantitative analysis of the speed/accuracy trade-off in transaction level modeling.
ACM Trans. Embed. Comput. Syst., 8(1):4:1–4:29, January 2009.

References III

- ▶ Anh T. Tran and Bevan Baas.
NoCTweak: A highly parameterizable simulator for early exploration of performance and energy of networks on-chip.
Technical Report ECE-VCL-2012-2, VLSI Computation Lab, ECE Department, University of California, Davis, 2012.

C++ String Implementation

- ▶ g++ 5.2.1
- ▶ for a 2 characters string:
 - ▶ stack space = 32, heap space = 0, capacity = 15
- ▶ for a 16 characters string:
 - ▶ stack space = 32, heap space = 17, capacity = 16
- ▶ 15 characters stack buffer to avoid dynamic memory allocation