InvadeSIM – A Simulator for Heterogeneous Multi-Processor Systems-on-Chip

Sascha Roloff, Frank Hannig, and Jürgen Teich
Hardware/Software Co-Design, Department of Computer Science
Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany

The trend of modern processor architectures to integrate a bunch of possibly heterogeneous processor cores into a single chip and to separate memory into peaces, which are joined to tiles connected via a network-on-chip (NoC), leads to a couple of challenges in programming such architectures with respect to non-functional objectives such as energy efficiency, resource utilization, and temperature distribution.

As a remedy, resource-aware programming concepts such as invasive computing have been proposed, which exploit self-adaptiveness and self-organization of thread generation and data distribution. Here, an application itself may decide which set of resources in terms of processor cores or memory is claimed in dependence on run-time status information of the resources (e. g., availability, utilization, temperature, load, etc.). According to the claimed set of resources, the workload is distributed among the tiles and threads are dynamically created for parallel processing. After this parallel execution, the claimed resources are freed again and become available for other applications.

In order to obtain important insights into the benefits of invasive computing, efficient simulations of the aforementioned platforms including architecture, runtime system and applications are mandatory. For this purpose, we present *InvadeSIM*, a parallel execution-driven simulator for functional and timing simulation of resource-aware applications on NoC-based MPSoCs (multi-processor system-on-chip) containing hundred to thousand heterogeneous cores. An overview of the InvadeSIM architecture is shown in Figure 1 and the special features of our simulator can be summarized as follows:

- Flexible modeling of heterogeneous tiled architectures in terms of number and type of tiles and processors on a tile such as standard RISC processors, cores with instruction set extensions, or dedicated hardware accelerators.
- Individual configuration options for processors, memory, and the NoC such as clock frequency, CPI, memory and cache access latency, flit size, buffer sizes, and transmission delays.
- Simulation of distributed parallel applications written in the modern programming language X10, which is based on a programming model called partitioned global address space (PGAS), where the address space is partitioned into places, which are mapped onto tiles of the MPSoC.
- Simulation of resource-aware programming concepts, which are embedded into the X10 programming language (InvadeX10 [1]) and realized by specially modeled hardware features and a resource management based on agents.
- Novel execution-driven simulation approach, where functional and timing simulation is tighly coupled [2]. Functional simulation is based on direct execution and measurements on the host machine, whereas timing simulation is based on an analytical approach called *time warping*.
- Different parallel discrete event simulation techniques to speed up simulation on a multi-core host machine [3].
- Cycle-accurate network-on-chip simulation for best effort and guaranteed service channels [4].
- Trace generation of the invasion and infection states of processors over time in a common trace file format.
- Qt-based live visualization GUI of the hardware and resource management states during simulation.

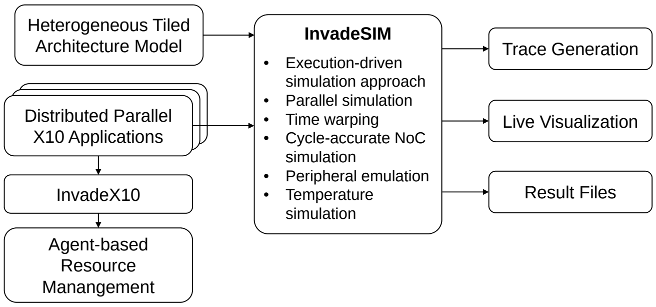
Heterogeneous Tiled Architecture Model
Distributed Parallel X10 Applications
InvadeX10
Agent-based Resource Manangement
InvadeSIM
Execution-driven simulation approach
Parallel simulation
Time warping
Cycle-accurate NoC simulation
Peripheral emulation
Temperature simulation
Trace Generation
Live Visualization
Result Files

Fig. 1. Overview of the components and their connections of the InvadeSIM heterogeneous MPSoC simulator.

Acknowledgment

This work was supported by the German Research Foundation (DFG) as part of the Transregional Collaborative Research Centre “Invasive Computing” (SFB/TR 89).

References

[1] F. Hannig, S. Roloff, G. Snelting, J. Teich, and A. Zwinkau, “Resource-aware programming and simulation of MPSoC architectures through extension of X10,” in *Proceedings of the 14th International Workshop on Software and Compilers for Embedded Systems (SCOPES)*, (St. Goar, Germany), ACM Press, Jun. 27–28, 2011, pp. 48–55.

[2] S. Roloff, F. Hannig, and J. Teich, “Approximate time functional simulation of resource-aware programming concepts for heterogeneous MPSoCs,” in *Proceedings of the 17th Asia and South Pacific Design Automation Conference (ASP-DAC)*, (Sydney, Australia), Jan. 30–Feb. 2, 2012, pp. 187–192.

[3] S. Roloff, D. Schafhauser, F. Hannig, and J. Teich, “Execution-driven parallel simulation of PGAS applications on heterogeneous tiled architectures,” in *Proceedings of the 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, (San Francisco, CA, USA), ACM, Jun. 7–11, 2015, 44:1–44:6.

[4] S. Roloff, A. Weichslgartner, J. Heisswolf, F. Hannig, and J. Teich, “NoC simulation in heterogeneous architectures for PGAS programming model,” in *Proceedings of the 16th International Workshop on Software and Compilers for Embedded Systems (M-SCOPES)*, (St. Goar, Germany), ACM, Jun. 19–21, 2013, pp. 77–85.