Autonomic Management of Reconfigurations in DPR FPGA-based Embedded System

Soguy Mak Karé Gueye, Eric Rutten, Jean-Philippe Diguet

Autonomic Management of Reconfigurations in DPR FPGA-based Embedded System

Soguy Mak-Karé Gueye, Éric Rutten
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG
F-38000 Grenoble, France
{soguy-mak-kare.gueye,eric.rutten}@inria.fr

Jean-Philippe Diguet
CNRS, Université Bretagne Sud, LAB-STICC
F-56321 Lorient, France
jean-philippe.diguet@univ-ubs.fr

INVITED TALK EXTENDED ABSTRACT

Implementing self-adaptive embedded systems, such as UAVs, involves an offline provisioning of the several implementations of the embedded functionalities with different characteristics in resource usage and performance in order for the system to dynamically adapt itself under uncertainties. FPGA-based architectures offer for support for high flexibility with dynamic reconfiguration features. We propose an autonomic control architecture for self-adaptive and self-reconfigurable FPGA-based embedded systems. The control architecture is structured in three layers: a mission manager, a reconfiguration manager and a scheduling manager. In this work we focus on the design of the reconfiguration manager. We propose a design approach using automata-based discrete control. It involves reactive programming that provides formal semantics, and discrete controller synthesis from declarative objectives.

A. The need for dynamically reconfigurable architectures

Like large-scale distributed systems, embedded systems such as Unmanned Vehicles (UV) are more and more required to be self-adaptive and self-reconfigurable, for resource management, energy efficiency, or by functionality. Many embedded systems, particularly embedded cameras, operate in dynamic, and often unpredictable environments so that a variety of complex tasks is required for a robust behavior of the system. Mission management and embedded intelligence also require online complex tasks. Context-aware and resource-aware adaptation by reorganizing the running tasks can lead to a better utilization of the system resources while retaining and possibly optimizing the performance and processing quality.

Field-programmable gate array (FPGA) devices are a promising solution for self-adaptive embedded systems. FPGAs allow to reach High Performances with the design of dedicated hardware implementations. Furthermore they also offer flexibility by means of Dynamic Partial Reconfiguration. DPR allows to track the best hardware implementation according to active task requirements. DPR FPGA supports virtually more hardware space for execution than statically available. Offline provisioning of several implementations of the tasks with different characteristics in resource usage (e.g., size and surface used) and performance (e.g., speed, quality) can be envisaged. All the tasks can not be active simultaneously due to area limitations. So when the context changes, adaptation and reconfiguration can be performed to select the appropriate subset of tasks suitable for the context; and run their compatible versions. As a result, the available resources can be optimally utilized under the control of reconfiguration managers which decide online on the moment when to switch, and on the choice of the next configuration to load according to mission requirements.

B. Model-based control of reconfigurations

We propose an autonomic control architecture [1] for self-adaptive and self-reconfigurable FPGA-based embedded systems [2]. The control architecture is layered so that the adaptation and reconfiguration decisions are taken at different levels. The architecture is structured in three layers. The higher-layer consists of a mission manager. The latter is responsible for adapting the system based on uncertainties in the environment. It determines the list of tasks that must be running. The middle-layer consists of a reconfiguration manager which receives from the mission manager the list of tasks to run. The reconfiguration manager is responsible for selecting the tasks implementations that satisfy the execution constraints specified by the mission manager; and that are compatible regarding the resources constraints. The lower-layer consists of a scheduling manager which receives the tasks implementations to run from the reconfiguration manager. The scheduling manager is responsible for processing the sequences of reconfigurations.

In this work we focus on the design of the reconfiguration manager. We build up on earlier work [3] where the multi-layered approach was not considered. Manual programming of such a manager could be error-prone, costly and complex due to the design space, namely the number of possible configurations to consider. Instead, we propose a design approach based on discrete control [4], supported by a reactive language [5]. The latter provides high level programming

languages for formal specification of possible configurations, tools such as Discrete Controller Synthesis; and powerful compilers automatically generating an executable implementation in C. This approach produces correct-by-construction controllers enforcing desired control objectives, and avoids error-prone manual programming and tedious debugging. We also involve a scheduling layer which executes the sequences of reconfigurations by generating a table encoding the scheduling process based on the tasks implementations to run.

C. Perspectives

The Perspectives are in several directions, amongst which we are currently implementing the case study of a video tracking system on a DPR FPGA, for which the bitstream implementations are ongoing. Moreover we are defining a DSL [6] allowing to describe architecture and application and objectives for automatic generation of the automata models and generation of the manager runtime code. We will exploit modularity supported by Heptagon/BZR [5] to control complex DPR FPGA architectures for scalability of design space exploration. Other perspectives are in co-ordination of multiple autonomic loops, switching controllers (integrating into the DSL schemes explored in another context [7]) ; modularity and hierarchical loops, both for re-use and to managed compilation complexity, and considering more ad-vanced discrete control features like logico-numeric properties [8] ; and integrating the DSL for Autonomic Managers into a global design process for FPGA-based applications. We also consider the general perspectives of basing Autonomic Management of Autonomic Computing systems on Control Theory [9,10].

Keywords : FPGA, Dynamic Partial Reconfiguration, Control, Performance, cost, and power management, FSM, Automata, Reactive Language

BIOGRAPHIES

ERIC RUTTEN, PhD 1990 and Hab. 1999 at University of Rennes, France, works at INRIA in Grenoble, in the field of Autonomic Computing. He currently works on the model-based control of adaptive and reconfigurable computing systems, using techniques from Control Theory, and ranging from embedded systems, to Cloud distributed systems and High-Performance Computing.

REFERENCES

[1] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer*, vol. 36, pp. 41–50, January 2003.

[2] S. M. K. Gueye, É. Rutten, and J. Diguet, "Autonomic management of missions and reconfigurations in fpga-based embedded system," in *2017 NASA/ESA Conference on Adaptive Hardware and Systems, AHS 2017, Pasadena, CA, USA, July 24-27, 2017*, 2017, pp. 48–55. [Online]. Available: https://doi.org/10.1109/AHS.2017.8046358

[3] X. An, E. Rutten, J.-P. Diguet, and A. Gamatié, "Model-based design of correct controllers for dynamically reconfigurable architectures," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 15, no. 3, Jun. 2016. [Online]. Available: http://dx.doi.org/10.1145/2873056

[4] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[5] G. Delaval, É. Rutten, and H. Marchand, "Integrating discrete controller synthesis into a reactive programming language compiler," *Discrete Event Dynamic Systems*, vol. 23, no. 4, pp. 385–418, 2013.

[6] S. M. K. Gueye , Gwenaël Delaval, É. Rutten, Dominique Heller and J.-P. Diguet, " A Domain-specific Language for Autonomic Managers in Hardware Reconfigurable Architectures" in Proc. Int. Conf. Autonomic Computing (ICAC 18), Trento, 2018.

[7] A. N. Sylla, M. Louvel, E. Rutten, and G. Delaval, "Design Framework for Reliable Multiple Autonomic Loops in Smart Environments," in *2017 IEEE International Conference on Cloud and Autonomic Computing (ICCAC)*, Tucson, AZ, United States, Sep. 2017.

[8] N. Berthier, F. Alvares, G. Delaval, H. Marchand, and E. Rutten, "Logico-numerical control for software components," in *1st IEEE Conference on Control Technology and Applications, CCTA*, 2017.

[9] M. Litoiu, M. Shaw, G. Tamura, N. M. Villegas, H. Müller, H. Giese, R. Rouvoy, and E. Rutten, "What Can Control Theory Teach Us About Assurances in Self-Adaptive Software Systems?" in *Software Engineering for Self-Adaptive Systems III. Assurances.*, ser. LNCS, R. de Lemos, D. Garlan, C. Ghezzi, and H. Giese, Eds. Springer, Jan. 2018, vol. 9640.

[10] E. Rutten, N. Marchand, and D. Simon, "Feedback Control as MAPE-K loop in Autonomic Computing," in *Software Engineering for Self-Adaptive Systems III. Assurances.*, ser. LNCS, R. de Lemos, D. Garlan, C. Ghezzi, and H. Giese, Eds. Springer, Jan. 2018, vol. 9640.