

System-Level Mapping, Analysis, and Management of Real-Time Applications in Many-Core Systems

Behnaz Pourmohseni, Jürgen Teich (Supervisor)

Department of Computer Science 12, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU)
Address: Cauerstr. 11, 91058 Erlangen, Germany. Email: {behnaz.pourmohseni, juergen.teich}@fau.de.

I. INTRODUCTION

Embedded domains of computing are undergoing a paradigm shift towards heterogeneous many-core platforms to empower the concurrent execution of an ever growing number of applications on a single chip. This shift is followed by a transition to dynamic application deployment and adaptive resource management schemes to enhance system utilization in view of the growing dynamism in the mix of running applications and the resulting fluctuations in resource availability and on-chip temperature distribution. In case of applications with hard real-time constraints, performance guarantees can be provided using a Hybrid Application Mapping (HAM) scheme, see Fig. 1. Here, a set of mapping alternatives with diverse resource requirements and satisfactory timing guarantees are computed using a Design Space Exploration (DSE) in an off-line characterization of individual applications. The mappings are then used for the on-line management of the system, e.g., to launch applications on demand or dynamically adapt the system based on precomputed mappings and in accordance with the current resource availability. The per-application characterization scheme in HAM necessitates system *composability* which ensures that concurrent applications cannot affect the non-functional properties of one another.

In this extended abstract, we introduce a collection of works which, for the first time, enable a fully composable and adaptable mapping and management of hard real-time applications. The outlined contributions are integrated into the standard flow of HAM (see Fig. 1) and propose novel techniques that enable a strictly predictable (red), thermally safe (green), scalable (orange), and adaptable (blue) deployment of hard real-time applications in heterogeneous many-core systems.

II. CONTRIBUTIONS

A. Isolation-Aware Characterization

Timing composability in many-core systems is typically realized by means of temporal/spatial isolation among applications, enforced by exclusive reservation of processing, communication, and storage resources (or resource budgets) per application at its launch time. In the current practice, each system is developed based on a *fixed* inter-application isolation scheme which imposes one fixed resource reservation policy on each and every application in the system, regardless of its compute-power and timing requirements. In [4], we demonstrated that a fixed isolation scheme often leads to sub-optimal mappings which suffer under-utilization of resources. As a remedy, we proposed in [4] an *isolation-aware application characterization* approach which introduces (i) *isolation-scheme exploration* and (ii) *isolation-aware timing analysis* into the off-line DSE in HAM (red boxes in Fig. 1). Here, the former explores various combinations of different isolation

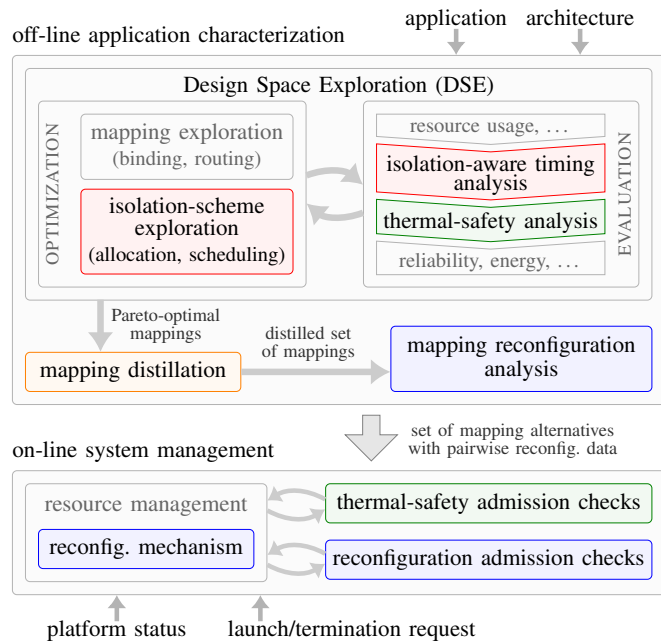


Fig. 1: Standard flow of Hybrid Application Mapping (HAM). The contributions outlined in this extended abstract enable a predictable (red), thermally safe (green), scalable (orange), and adaptable (blue) deployment of applications with hard real-time requirements in heterogeneous multi-/many-core systems.

schemes within each computed mapping while the latter derives worst-case timing guarantees for the thusly obtained mappings. Contrarily to existing timing analyses, the proposed analysis is applicable to mappings with arbitrary combinations of isolation schemes and delivers safe yet tight worst-case timing bounds per mapping by identifying and excluding inter-application interference scenarios that are impossible under the selected mix of isolation schemes. Experimental results demonstrate that the proposed approach achieves an average improvement of 26% (up to 67%) in the quality of obtained mappings, compared to fixed-isolation-scheme approaches.

B. Thermal Safety and Composability

The massive integration of resources in many-core platforms comes at the cost of a significant increase in the on-chip power density which may often engender hot spots across the chip and even jeopardize its thermal integrity. To ensure thermal safety, many-core platforms employ Dynamic Thermal Management (DTM) techniques which continuously monitor the thermal state of the chip and take countermeasures such as power gating at the sight of thermal violations, which, in turn, affects the timing behavior of the applications running in those hot spots. The current practice of HAM disregards thermal safety. Consequently, it not only fails in ensuring a seamless satisfaction of real-time requirements, but also cannot preserve composability since,

the thermally unsafe behavior of one application may lead to interferences with the intended behavior of other applications.

As a remedy, in [7] we proposed a *thermally composable HAM* methodology which enforces thermal safety proactively (preventing thermal violations) and preserves composability by eliminating DTM interferences. Our approach comprises (i) a *thermal-safety analysis* and (ii) a set of lightweight *thermal-safety admission checks* (green boxes in Fig. 1). The former is integrated into the off-line DSE and derives a thermal-safety label for each computed mapping while the latter is employed on-line at the launch time of applications to verify, for the current system utilization, the thermal safety of mapping candidates based on their precomputed labels. This allows identifying and avoiding mappings that may induce thermal violations, which in turn, eliminates the exposure of real-time applications to DTM countermeasures without compromising the thermal integrity of the chip. We experimentally verified in [7] the effectiveness of the proposed methodology in establishing thermal safety proactively and, by that, preserving composability.

C. Deployment Adaptation

Embedded many-core systems manifest a highly dynamic environment where performance requirements of applications may change dynamically due to workload variations, and resources may become unavailable unexpectedly due to hot spots or hardware faults. In this context, deployment adaptability becomes necessary to enable reacting to such unforeseeable events. Chiefly relying on task migration, deployment adaptation may become a source of unpredictability which is unacceptable for hard real-time applications. Moreover, the timing verification of the post-adaptation deployment often relies on compute-intensive analyses that are not suitable for on-line use.

To enable an adaptable deployment with hard real-time guarantees and affordable compute overhead, we presented in [2] a *real-time mapping reconfiguration* methodology. There, we investigated deployment adaptations in the form of a transition between the statically computed set of mapping alternatives of each real-time application. This approach, on the one hand, eliminates the need for an on-line timing analysis of the post-adaptation deployment, since the worst-case timing behavior of the mappings is already analyzed during the DSE. On the other hand, since the set of mappings is known statically, the worst-case timing overhead for switching between each pair of mappings can be analyzed statically and used on-line. To that end, in [2] we presented (i) a *deterministic reconfiguration mechanism* based on which we proposed (ii) an *off-line reconfiguration analysis* (see the blue boxes in Fig. 1). Here, the former enables performing mapping reconfigurations in a predictable fashion while the latter bounds the worst-case latency of each reconfiguration. In our reconfiguration analysis, we first identify efficient migration routes with minimized allocation overhead and migration latency for each pair of mappings. Then, we formally bound the worst-case latency of the whole reconfiguration process according to the timing characteristics of the source and target mappings and the identified migration route for each migrating task. At run time, the real-time conformity of each reconfiguration candidate is verified using (iii) *reconfiguration admission checks*. In [3], we extended our off-line analysis and presented a (iv) *hybrid reconfiguration analysis* which performs the compute-intensive optimization of migration routes off-line while postponing parts of the latency analysis to run time, enabling taking the *actual* system load into account to tighten the reconfiguration latency bounds. Our experimental results in [2] demonstrated

the effectiveness of mapping reconfiguration in enabling hard real-time deployment adaptations. Moreover, we showcased in [3] the advantage of our hybrid reconfiguration analysis over its off-line counterpart in case study on thermal management of real-time many-core systems using mapping reconfiguration.

D. Mapping Distillation

The ever growing number of non-functional properties to be regarded during the off-line characterization of applications introduces a large number of design objectives in the DSE: Mappings must be optimized w.r.t. multiple quality objectives, e.g., worst-case latency, throughput, thermal safety, and energy. Moreover, to obtain mappings that support various run-time resource availability scenarios in heterogeneous systems, the number of cores of each type required by each mapping must be minimized, resulting in one design objective per core type. However, this large number of design objectives often results in a huge set of Pareto-optimal mappings which imposes an unacceptable management and storage overhead for the run-time system. To alleviate this overhead, in [1] we introduced a *mapping distillation methodology* to select a subset of the mappings for on-line use such that an acceptable run-time management and storage overhead is achieved while keeping the negative impact of this reduction to a minimum (orange box in Fig. 1). To that end, we first divide the space of non-resource objectives into regions of comparable trade-offs using a hyper-grid-based approach. Subsequently, resource-efficient mappings are distilled from each quality region using Pareto ranking in the space of resource objectives. We experimentally showed that our approach significantly enhances the application's run-time embeddability and delivers competitive and often superior quality properties, compared to existing truncation approaches. A copy of [1] is submitted along with the extended abstract at hand.

III. SUMMARY

The contributions outlined in Section II enable the efficient design and adaptive management of real-time applications in many-core systems. These and the contributions developed thereupon have been published in distinguished conferences and renowned journals from the areas of design automation and real-time systems, a selection of which is listed in the references.

REFERENCES

- [1] B. Pourmohseni, M. Glaß, and J. Teich, "Automatic operating point distillation for hybrid mapping methodologies," in *Proc. Design, Automation, and Test in Europe (DATE)*, 2017, pp. 1135–1140.
- [2] B. Pourmohseni, S. Wildermann *et al.*, "Predictable run-time mapping reconfiguration for real-time applications on many-core systems," in *Proc. Real-Time Networks and Systems (RTNS)*, 2017, pp. 148–157, **Outstanding Paper Award. Best Presentation Award.**
- [3] B. Pourmohseni, S. Wildermann *et al.*, "Hard real-time application mapping reconfiguration for NoC-based many-core systems," *Real-Time Systems (RTS)*, vol. 55, no. 2, pp. 433–469, Springer, 2019.
- [4] B. Pourmohseni, F. Smirnov *et al.*, "Isolation-aware timing analysis and design space exploration for predictable and composable many-core systems," in *Proc. Euromicro Conf. Real-Time Systems (ECRTS)*, 2019, pp. 12:1–24.
- [5] F. Smirnov, B. Pourmohseni *et al.*, "Variety-aware routing encoding for efficient design space exploration of automotive communication networks," in *Vehicle Technology and Intelligent Transport Systems (VEHITS)*, 2019.
- [6] F. Smirnov, B. Pourmohseni *et al.*, "Igor, get me the optimum! prioritizing important design decisions during the DSE of embedded systems," *ACM Trans. Embedded Computing Systems (TECS)*, vol. 18, no. 5s, p. 78, 2019.
- [7] B. Pourmohseni, F. Smirnov *et al.*, "Thermally composable hybrid application mapping for real-time applications in heterogeneous many-core systems," to appear in *Proc. Real-Time Systems Symposium (RTSS)*, 2019.
- [8] J. Teich, B. Pourmohseni *et al.*, "Run-time enforcement of non-functional requirements in heterogeneous many-core systems," to appear in *Proc. Asia and South Pacific Design Autom. Conf. (ASP-DAC)*, 2020.
- [9] F. Smirnov, B. Pourmohseni, and J. Teich, "Using learning classifier systems for the DSE of dynamically adaptable embedded systems," to appear in *Proc. Design, Automation, and Test in Europe (DATE)*, 2020.