Hyperreconfigurable Systems – Formal concepts, architectures and applications

by Sebastian Lange

Dynamically reconfigurable hardware offers promising possibilities for flexible, computation intensive applications. With the technological advance of reconfigurable hardware came a rapid growth in the number of resources per chip requiring large amounts of data transfer per reconfiguration operation. Especially run-time reconfigurable applications, which make frequent use of reconfiguration, suffer from the induced growing overhead.

In this thesis, we investigate concepts for reconfigurable architectures that can dynamically reconfigure their reconfiguration potential to adapt to the needs of a computation. Such architectures are called hyperreconfigurable. This new concept is promising, because hardware algorithms typically show phases with differing requirements on resources. First, we investigated a 2-level model of reconfiguration that uses ordinary reconfiguration operations and higher-level hyperreconfiguration operations. The hyperreconfiguration operations change the architecture's ability for reconfiguration and lower level ordinary reconfigurations specify the contexts for a computation within the limits set by the preceding hyperreconfiguration. Formal models for fine- and coarse-grained hyperreconfigurable architectures allow an objective evaluation of the capacity of hyperreconfigurable systems for reducing the reconfiguration overhead. A fundamental problem for such architectures is how to decide optimally when and which hyperreconfiguration operations should be performed during a computation in order to minimize the total reconfiguration time (PHC-problem). Although the general problem is NP-hard, fast polynomial time algorithms for solving this problem on interesting special types of hyperreconfigurable systems were developed. In order to test the formal models, the example Simple Hyperreconfigurable Architecture (SHyRA) was introduced. The reconfigurable resources of SHyRA closely resemble a one-dimensional FPGA with a minimum of extra facilities necessary for storing the hyperreconfiguration data. Several test applications were mapped onto the architecture using hyperreconfigurations. They show a decrease in reconfiguration overhead of more than 25% - 80% when compared to ordinary reconfiguration. A further reduction of the reconfiguration overhead was could be observed when introducing a cache architecture for storing data hypercontext data within the system. It was studied, how systems can benefit from potential synergetic effects of hyperreconfiguration and the use of caches and efficient heuristic solutions for the solution of the PHC problem were established.

The concept was further extended to an arbitrary number of levels, where each level restricts the potential for reconfiguration of its lower levels. We presented fast polynomial algorithms for the PHC-problem on such systems and showed that the use of more than two levels of reconfiguration can further decrease the reconfiguration overhead. Hyperreconfigurable systems need to provide additional hardware resources for storing and processing the hypercontext data. The optimal granularity of these operations was investigated with respect to the performance and the amount of additional hardware resources necessary. As fine granularity allows for more flexibility and coarser granularity requires fewer resources, good heuristic solutions for striking the balance were provided. The results were used to explore the design space of hyperreconfigurable systems with respect to optimizing the number of reconfiguration levels and granularity of reconfiguration.

The concluding part of this thesis explores the concept of dynamic partial reconfiguration, used in commercially available FPGAs. These systems were formally modeled, which provided insight into parameters and key issues of the concept, such as the optimal granularity of reconfiguration, the limits on its usefulness and the applicability of diverse granularity. A conceptual comparison of hyperreconfiguration and dynamic partial reconfiguration shows that hyperreconfiguration can improve on the results of the existing approach.