Multi-Objective Design Space Exploration of Embedded Systems in a Grid Environment

Vincenzo Catania, Gianmarco De Francisci Morales, Alessandro G. Di Nuovo, Maurizio Palesi, Davide Patti

Dipartimento di Ingegneria Informatica e delle Telecomunicazioni, Università degli Studi di Catania, Italy

Abstract

Design Space Exploration (DSE) is a hot issue in embedded system design. One of the main difficulties which characterize DSE problems is related to the computational power requirements for implementing any reliable DSE strategy. To be reliable, a DSE strategy has to operate at the appropriate abstraction level. It cannot be too high as important details which affect the objectives to be optimized might be not modeled appropriately. On the other hand, operating at a low abstraction level, could make it unreliable to evaluate the objectives for large real life data sets. The use of system-level simulators built around an instruction-set simulator is considered as the most viable solution in trading-off both the aspects of accuracy and efficiency. Although they allow to evaluate many system configuration running a given application in a reasonable amount of time, they are still too expensive to be used as a core of a DSE strategy which requires the simulation of thousands of system configurations. In this paper we assess the use of High Performance Computing (HPC) in DSE of a complex highly parameterized VLW based System-on-a-Chip (SoC) platform. Experiments show that the conventional wisdom of linear decrease in exploration time as the number of available processors increases is violated starting from a relatively low number of processors mainly due to communication overhead and I/O bottleneck.

Tests and Results

The testing was done using the following configuration: 128x128 LSU16 Black with 2 Octagons 2 Ghz dual core processors (for a total of 454 cores per blade) equipped with 1088 of DDR and a 153GB SATA HDD by SiliconStripes. The cluster used for the tests was configured to allow 128 processes per processor, and one of the hosts is reserved to run cluster services and to manage the jobs, so the maximum reachable number of MP processes was 50. The software package was installed on the dedicated blade that was also the cluster coordinator. The package was retrieved via the batch queue manager on a per-job basis on all the hosts involved in the parallel computation. The tests were done with no interference from other jobs in the cluster, so the scheduler tried to allocate the processes as close as possible in order to be one host before employing the next. This is an advantage because it reduces the number of copies of the software to be made.

Run the simulation of 1000 configurations using a growing number of processors, and even using an exponential growing number of processors the wall clock time was reduced only by an order of magnitude in the environment increases with the number of hosts. In our HPC environment each host has its own storage unit, and the simulation environment has to be copied. For this reason, it is important to have a good strategy for the distribution and management of the data, in order to provide data transfer and management, and for the allocation and distribution of environment data.

Conclusions

In this paper we presented a case study of design space exploration of a complex highly parameterized VLW based SoC platform. The 18 free parameters which characterize the platform span a design space of over 10^14 system configurations. Even considering an evaluation time of a few seconds for each configuration, exhaustive exploration would take hundreds of years on single machine. We test the use of High Performance Computing (HPC) as a viable solution to tackle with this problem. However, a maximum reduction of an order of magnitude in exploration time has been observed in our experiments. Meanwhile, statistical or machine learning approaches to the DSE problem presented in the literature, achieved average savings of two order of magnitude than classical approaches. This result reveals that the use of a combination of statistical and machine learning approaches in HPC environments is likely to be the most promising way to significantly reduce the exploration time.