The BarbequeRTRM
A run-time resource manager for multi/many-core architectures

Patrick Bellasi, Giuseppe Massari and William Fornaciari
Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy

Problem

Modern and future many-core platforms are going to offer considerable computational capabilities. Current parallel programming models (e.g. OpenCL, OpenMP, ...) aim to exploit such architectures focusing on the performance maximization. Mobile and embedded systems needs applications that could adapt their level of parallelism according to changes in resource availability, due to contention, power budgets and response to critical thermal conditions. Therefore, efficient run-time resource management (RTRM) techniques [1] are a key goal in the usage of these platforms. The BarbequeRTRM framework proposes a solutions that overcome the lack of general-purpose operating systems, performing a multi-objective resource partitioning and supporting reconfigurability of run-time managed applications.

Run-Time Resource Management

The BarbequeRTRM is a framework for adaptive RTRM of many-core architectures, offering optimal resource partitioning and adaptive run-time scheduling of the different reconfigurable applications. It has been designed to be the core of a highly modular and extensible run-time resource manager, providing support for an easy integration and management of multiple applications competing on the usage of one (or more) shared MIMD many-core computation devices. It is possible to support both homogeneous and heterogeneous architectures, since the the low-level communication (Platform Proxy) is handled by a system specific module.

Management policies optimize resource assignment to demanding applications considering:
- application properties, e.g., resource requirements, Quality-of-Service and relative priorities
- resources availability and state, e.g. frequency, power consumption, process variation and thermal conditions.
- optimization goals, e.g., power reduction, energy optimization, reconfiguration overheads minimization, congestion avoidance, fairness and overall performances maximization.

Application integration and Results

The framework provides a library (RTLib) for the integration of the application with the RTRM. In the RTLib a base C++ class is defined (BbqueEXC) exporting a small set of callback methods.

From the RTRM point of view, each independent module of the application represents an Execution Context (EXC), and must be implemented as a class derived from BbqueEXC. Therefore the RTRM manages the execution flow of the EXCs, by invoking the methods according to the Abstract Execution Model (AEM).

The AEM allows the EXC to be aware of changes of resource availability. Considering a typical parallel application, this means to adapt its level of parallelism, i.e., number of running threads, or OpenCL kernels; along with redefine usage of memory by resizing the data buffers, and so on.

References