Generalized strictly periodic scheduling analysis, resource optimization, and implementation of adaptive streaming applications

Niknam, S.

Author: Niknam, S.
Title: Generalized strictly periodic scheduling analysis, resource optimization, and implementation of adaptive streaming applications
Issue Date: 2020-08-25

Summary

This thesis focuses on addressing four research problems in designing embedded streaming systems. Embedded streaming systems are those systems that process a stream of input data coming from the environment and generate a stream of output data going into the environment. For many embedded streaming systems, the timing is a critical design requirement, in which the correct behavior depends on both the correctness of output data and on the time at which the data is produced. An embedded streaming system subjected to such a timing requirement is called a real-time system. Some examples of real-time embedded streaming systems can be found in various autonomous mobile systems, such as planes, self-driving cars, and drones.

To handle the tight timing requirements of such real-time embedded streaming systems, modern embedded systems have been equipped with hardware platforms, the so-called Multi-Processor Systems-on-Chip (MPSoC), that contain multiple processors, memories, interconnections, and other hardware peripherals on a single chip, to benefit from parallel execution. To efficiently exploit the computational capacity of an MPSoC platform, a streaming application which is going to be executed on the MPSoC platform must be expressed primarily in a parallel fashion, i.e., the application is represented as a set of parallel executing and communicating tasks. Then, the main challenge is how to schedule the tasks spatially, i.e., task mapping, and temporally, i.e., task scheduling, on the MPSoC platform such that all timing requirements are satisfied while making efficient utilization of available resources (e.g, processors, memory, energy, etc.) on the platform. Another challenge is how to implement and run the mapped and scheduled application tasks on the MPSoC platform. This thesis proposes several techniques to address the aforementioned two challenges.

In the first part of the thesis, the focus is on addressing the first aforementioned challenge in the design of embedded streaming systems. To do so, a scheduling framework is proposed to convert the data-dependent tasks in an application, including cyclic data-dependent tasks, to real-time periodic

tasks. As a result, a variety of hard real-time scheduling algorithms for periodic tasks, from the classical real-time scheduling theory, can be applied to schedule such streaming applications with a certain guaranteed performance, i.e., throughput/latency. These algorithms can perform fast admission control and scheduling decisions for new incoming applications in an MPSoC platform as well as offer properties such as temporal isolation and fast analytical calculation of the minimum number of processors needed to schedule the tasks in the application.

In the second part of the thesis, the focus is on addressing the problem of efficiently exploiting resources on an underlying MPSoC platform when scheduling the tasks of applications on the platform. An algorithm is proposed to transform an initial representation of a streaming application, i.e., an initial application graph, into a functionally equivalent one such that the new representation requires fewer processors while guaranteeing a given throughput requirement. Additionally, this thesis studies the problem of energy-efficient scheduling of streaming applications with throughput requirements on MPSoC platforms with voltage and frequency scaling capability. In this regard, a novel periodic scheduling framework is proposed which allows streaming applications to switch their execution periodically between a few energy-efficient schedules at run-time in order to meet a throughput requirement at long run. Using such periodic switching scheme, system designers can benefit from adopting Dynamic Voltage and Frequency Scaling techniques to exploit available static slack time in the schedule of an application efficiently.

Finally, in the third part of the thesis, the focus is on addressing the second aforementioned challenge in the design of embedded streaming systems. In this regarde, a generic parallel implementation and execution approach for (adaptive) streaming applications is proposed. The proposed approach can be easily realized on top of existing operating systems while supporting the utilization of a wider range of schedules. In particular, a demonstration of the proposed approach on LITMUS^{RT} is provided, which is one of the existing real-time extensions of the Linux kernel.