QUAD - Quantitative Usage Analysis of Data

S. Arash Ostadzadeh, Roel J. Meeuws, Carlo Galuzzi and Koen Bertels
Computer Engineering Group
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, Delft, the Netherlands
{S.A.Ostadzadeh,R.J.Meeuws,C.Galuzzi,K.L.M.Bertels}@tudelft.nl

Abstract. In recent years, the shift towards multi-core systems introduced new challenges that need to be taken into consideration to fully exploit the capabilities of these heterogeneous systems. In particular, there is a compelling need for utility tools to facilitate the application development process, tuning and optimization. Tools to understand the memory access behavior of an application are inevitably vital for optimizing the execution of data-intensive streaming applications on heterogeneous architectures, as improvements in processing performance continue to outpace improvements in memory performance.

In this poster, we present the *Quantitative Usage Analysis of Data* (QUAD) tool, a sophisticated memory access tracing tool that provides a comprehensive quantitative analysis of memory access patterns of an application. The primary goal of this tool is to detect the actual data dependencies at function-level. This dynamic binary analysis tool allows automatic runtime detection of actual data dependency between functions in contrast to conventional data dependency discovered by similar memory access analysers. Although QUAD can be employed to spot coarse-grained parallelism opportunities in an application, it practically provides a more general-purpose framework that can be utilized in various optimizations by estimating effective memory access related parameters. Additionally, QUAD does not require any modification of the application binaries and it has no compiler dependence other than debug information. It also abstracts away from the properties of a particular architecture.