Abstract

Raghda H El-Shehaby

Empirical Analysis of Graphics Processing Units Architectures (GPUs): Study of the Performance of GPUs

Graphics Processing Units (GPUs) have the capability of very high speedup over conventional processor for highly-parallel general-purpose computations. Different hardware gives different performance and efficiency. The manufacturers of these devices provide little information about the characteristics of their hardware that being so, understanding GPU performance is still a complicated process. Following a performance model helps identify the parameters to be measured, then microbenchmarks allow the empirical measurement of these characteristics. The use of a model also explains the behavior of the hardware while running a specific application. Eventually, the model provides an estimate of the performance. The GPU computing lab of the Vrije Universiteit Brussel (VUB) department of Electronics and Informatics (ETRO) is currently working to develop a theoretical model to estimate the performance of any given GPU, called the Pipeline performance model. The model defines a set of lambda parameters, issue latency (?) and completion latency (?), and additionally characterizes the hardware with concurrent execution constraints. The work in this thesis seeks to expose each characteristic, by designing and creating a set of microbenchmarks for various types of operations (single precision, double precision, special function, .etc). These microbenchmarks are run over different GPU generations, to capture the values of the parameters. These experiments aim to validate the Pipeline model, by comparing the results to theoretical findings, and test its accuracy for different GPU architectures. The results obtained successfully proved the efficiency of the Pipeline model. The practical lambdas were found to almost correctly match theoretical calculations. The lambdas were measured for single precision, double precision, special functions and a mix of these operations. The use of benchmarks correctly identified the concurrent execution constraints in the different architectures. Finally, one of the approaches used to extend the model to support various types of operations, the weighted average method, was chosen because of its ability to represent the hardware to a great degree of accuracy.