# 論 文 の 内 容 の 要 旨

論文題目　A Framework for Performance Analysis and Optimization for GPU Kernel Programs using Linear Performance-Breakdown Model
（線形性能分解モデルを用いたGPUカーネルプログラムの性能解析と最適化のためのフレームワーク）

氏　　名　　チャパ　マルテル　マリオ　アルベルト

This thesis presents a performance modeling work for Graphic Processing Units, the Performance-Breakdown Model LBPM. GPUs are nowadays a commonly used accelerator in computer systems because its high degree or parallelism and high arithmetic throughput capabilities have proven useful for many applications outside the realm of 2D and 3D graphics processing. From commercial, desktop computers to supercomputer systems, include in their design GPUs as computing accelerator devices. While it is true that the benefits of using a GPU as an accelerator device are significant, the effort and time required to produce a program that makes optimal use of the GPU is also high, which represents of the major drawback of the General Purpose GPU technology.

GPUs are highly sophisticate hardware designed to create and manage thousands of parallel processes which is the source of their high computational throughput, nonetheless this sophistication is carried out at the expense of great architectural complexity.

The complexity of the GPU architecture is the reason why it is difficult and time-consuming for programmers to achieve the high performance that GPUs can attain. Although there have been much effort devoted to application development and algorithms optimization in the GPU community, the effort devoted to performance modeling and analysis tools is not as prominent. Whilst there are

several works on the literature regarding performance models for GPU, it is worth mentioning that a large share of them focus on the CUDA programing model and hence applicable only to Nvidia GPUs. Furthermore, many of them make use of hardware performance counters and specific characteristics of the GPU architecture. The drawback of such approach is that the models and tools can only be applied to Nvidia GPUs and there is no warranty that they might be valid for future generations hardware. However, this studies provide insight about the inner workings of Nvidia GPUs that serve as a base for understanding the general architecture of the GPU device, allowing us to extract this insights and applying it to our own model.

We recognize the need of attacking the problem of the difficult and time-consuming task of producing optimal GPU kernel programs by developing a tool that can be used as a guide to understand the behavior of an application when executing in GPU systems. We chose to use the OpenCL programming model for its universality; a kernel program written using the OpenCL standard can be executed in any compliant GPU, independently of the vendor or the device's family. Our model do not require use of internal performance counters or other device-specific metrics, which improves its usability.

The proposed system features the following sub-systems:

[Auto-Profiling Module (APM)] This module generates a set of runs to collects profiling data (execution time against work-group size). It makes use of both, kernel and host code files.

[CLParser Module (CLPM)] Takes the host and kernel program codes and generates an Abstract Syntax Tree for each. The trees are transversed in order to locate the key control loops, variables and function calls; information used to discover movement of data between the different levels of the memory hierarchy.

[Linear Regression Module (LRM)] This module uses the profiling information and the performance formula to calculate the Regression Coefficients by applying the Least Squares Method with non-negativity constraint.

[Performance Breakdown Module (PBM)] The last module calculates the

performance-breakdown using the Regression Coefficients and the performance formula.

In general, the input to the system is the files containing the host and kernel code. The output after being processed by the different modules will be a graph showing the performance breakdown.

The main contributions of this thesis include:

The Linear Performance-Breakdown Model

Semi-automatic performance model generation

Integration into a script-based framework

we demonstrated the capabilities of the model using two different GPU devices: an AMD's APU (integrated GPU device) and a Nvidia Geforce GTX 660 (discrete board) using two different kernel programs, GSMM and FFT . Both of the used devices have a different architecture, one being a Nvidia card and the other being a AMD card, additionally to the fact that the GTX 660 is a discrete card and the APU is an integrated device. This shows that the model is capable of capture the hardware characteristics of different types of GPU and offer an accurate breakdown model.