# Estimating the Execution Time of CUDA Programs

**Gabriel WARDE**

CRIStAL Laboratory, Lille, France

**Supervised by:**

Pr. Giuseppe LIPARI
Nordine FEDDALL

December 4, 2024

**Abstract**

The growing use of GPUs in critical systems, such as autonomous driving and scientific computing, demands accurate execution time estimation for CUDA programs. This report examines the challenges of modeling GPU behavior, considering the complexity of architectures and diverse kernel configurations. Analytical methods offer simplicity, while machine learning provides adaptability, each with trade-offs. By reviewing state-of-the-art approaches and identifying gaps, this work seeks to enhance models for better accuracy, portability, and scalability in GPU applications.

CRIStAL Centre de Recherche en Informatique, Signal et Automatique de Lille
Université de Lille
centralelille ÉCOLE CENTRALE DE LILLE
IMT Atlantique Bretagne-Pays de la Loire École Mines-Télécom

# Contents

# 1. Introduction

## 1.1. Motivation

Modern computing systems rely heavily on GPU acceleration for high-performance tasks like deep learning, computer vision, and scientific simulations. Accurately estimating the execution time of CUDA processes is crucial for optimizing resource usage and ensuring real-time performance in critical applications such as autonomous driving.

## 1.2. Objective

This report explores the current state-of-the-art methods for predicting the execution time of GPU kernels, focusing on analytical models and machine learning (ML) approaches. The goal is to situate our research within existing literature and try to identify opportunities for improvement.

# 2. Related Work and Analysis

## 2.1. Paper 1: *Evaluating Execution Time Predictions on GPU Kernels Using an Analytical Model and Machine Learning Techniques* [1]

### 2.1.1 Findings

This study compares Bulk Synchronous Parallel (BSP)-based analytical models with machine learning approaches like Linear Regression, SVM, and Random Forest. ML techniques with feature extraction achieved errors as low as 1.54% on unseen GPUs.

### 2.1.2 Critique

While effective, the reliance on extensive feature extraction limits its scalability to new datasets. The analytical model's utility in scenarios with irregular kernel behavior is also limited.

### 2.1.3 Relevance to Our Work

The emphasis on feature extraction aligns with our aim to explore feature-efficient ML methods for CUDA kernel predictions.

## 2.2. Paper 2: *A Simple Model for Portable and Fast Prediction of Execution Time and Power Consumption of GPU Kernels* [2]

### 2.2.1 Findings

This paper presents a lightweight random forest-based model using hardware-independent features. It achieves portable predictions across GPU architectures with a median MAPE

of between 8.86% and 13.86% for server-class GPUs and 52% for the consumer-class GTX1650.

### 2.2.2 Critique

The simplicity of the model, while beneficial for portability, may limit its accuracy on highly optimized or adversarial kernels.

### 2.2.3 Relevance to Our Work

Its focus on hardware-independent features resonates with our goal of developing scalable and adaptable ML models.

# 3. Conclusion and Future Work

We have synthesized findings from existing literature to establish a foundation for our research. Future steps include effective CUDA metrics/feature extraction from the ncu command, model development (Machine Learning and Deep Learning techniques), experimental validation, and benchmarking against state-of-the-art approaches.

# References

[1] Marcos Amaris, Raphael Camargo, Daniel Cordeiro, Alfredo Goldman, and Denis Trystram. Evaluating execution time predictions on gpu kernels using an analytical model and machine learning techniques. *Journal of Parallel and Distributed Computing*, 171:66–78, 2023.

[2] Lorenz Braun, Sotirios Nikas, Chen Song, Vincent Heuveline, and Holger Fröning. A simple model for portable and fast prediction of execution time and power consumption of gpu kernels. *ACM Trans. Archit. Code Optim.*, 18(1), December 2021.