Instrumentation and Modeling of Performance and Power Consumption for Massively Parallel Processors

Chen Song

Heidelberg University
chen.song@iwr.uni-heidelberg.de

19/01/2021

IWR
Interdisciplinary Center
for Scientific Computing

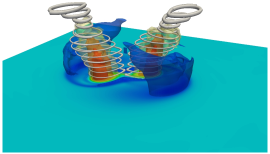
ziti

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

Bundesministerium
für Bildung
und Forschung

GPU Mekong Project - Simplified Multi-GPU Programming

- Aim & Objective: provide a simplified path to scale out the execution of GPU programs from one GPU to almost any number.
- Funding: Federal Ministry of Education and Research of Germany - BMBF.
- Funding period: 2017.02. – 2020.06.
- Host Institute: Heidelberg University, Germany.
  - Engineering Mathematics and Computing Lab (EMCL), Mathematics Faculty.
  - Computing Systems Group (CSG), Informatics Faculty.
- The name "Mekong".
- Project website: https://www.gpumekong.org/



Research Team

Engineering Mathematics and Computing Lab (EMCL)

Vincent Heuveline

Chen Song

Sotirios Nikas

Simon Gawlok

Computing Systems Group (CSG)

Holger Fröning

Lorenz Braun

Alexander Matz

Highlight Developments within GPU Mekong Project

- Mini-Apps:
  - Finite Element method (FEM) based CPU-GPU benchmark suites.
  - Various solvers and schemes: e.g. CG, GMRES, Multi-Grid, Matrix-Free, ...
  - https://emcl-gitlab.iwr.uni-heidelberg.de/mini_apps/Mini-Apps_Public
- CUDA Flux:
  - Lightweight instruction profiler for CUDA applications.
  - PTX level.
  - LLVM compiler framework based.
  - Low Overhead.
  - https://github.com/UniHD-CEG/cuda-flux
- GPU Mangrove:
  - Performance & Power prediction model.
  - Fast and easy to use.
  - Machine learning based.
  - https://github.com/UniHD-CEG/gpu-mangrove

HiPEAC Tutorial

- Background:
  - GPU application: typical example for heterogeneous computing.
  - Predictive model can assist the scheduler.
  - **Performance** and **Power** are two main metrics for designing algorithms and compute architecture.
- Our predictive model:
  - **Simple**: only rely on features obtained with minimal overhead.
  - **Portable**: easily transported to other GPU architectures.
  - **Fast**: machine learning based model, computing time is limited.
- Toady's tutorial main content:
  - Instrumentation.
  - Predictive model for performance and power.
- Length: full day.
  - Morning: Background and methodology.
  - Afternoon: Tooling and hands-on experiments.
- Publication:
  - **A simple model for portable and fast prediction of execution time and power consumption of gpu kernels**, *ACM Trans. Archit. Code Optim.* Dec. 2020.

Program

09:30 – 09:40	Introduction	Chen Song
09:40 – 10:00	General Introduction for GPU	Holger Fröning
10:00 – 10:30	Instrumentation in general	Lorenz Braun
10:30 – 11:15	Break	
11:15 – 11:45	Instrumentation for performance & power	Lorenz Braun
11:45 – 12:15	Building predictive models	Lorenz Braun
12:15 – 12:45	Cluster, tools and exercise introduction	Yannic Emonds
12:45 – 15:00	Lunch & Keynote	
15:00 – 16:00	Exercise - performance & power measurements	L. Braun & Y. Emonds
16:00 – 16:30	Break	
16:30 – 17:30	Prediction experiments	Hands-on
17:30 – 18:00	Summary predictions & wrap-up	Lorenz Braun

Thanks for your attention

Enjoy our tutorial