Università
della
Svizzera
italiana

Faculty
of Informatics

USI Technical Report Series in Informatics

Particle Simulations with OpenACC: Speedup and Scaling

Samuel A. Cruz Alegría, Alessandra M. de Felice, Hrishikesh R. Gupta[1]

[1] Faculty of Informatics, Università della Svizzera italiana, Switzerland

Abstract

The simulation of particle systems has become essential for visualizing the behaviour of relevant physical systems, ranging from simulations of molecular dynamics to simulations of colliding galaxies. The computational complexity of performing simulations grows with the number of particles in the system. Performing realistic simulations may necessitate a plethora of particles, leading to immense computational costs. Simulating such systems may thus require increasingly longer time frames. Hence, performing increasingly complex simulations may become impractical for single-core simulation tools. Thus, it is essential to develop simulation tools which perform practically independent of the number of bodies used in a simulation. A possibility to reduce the time required for simulations is to distribute the workload among different parallel entities, such as different processes or threads. This paper aims to explore the efficiency and scalability of parallelization in order to improve the performance of a simulation currently run on a single core. This is achieved by incorporating the OpenACC programming standard, which is a programming standard for parallel computing that utilizes a hardware accelerator, such as a GPU.

Report Info

Published
June 2018

Number
USI-INF-TR-2013-3

Institution
Faculty of Informatics
Università della Svizzera italiana
Lugano, Switzerland

Online Access
www.inf.usi.ch/techreports

1 Introduction

1.1 Particle Simulations

- What are they?
- Why are they important?
- How can we model this?
- What does our study deal with?
- What are assumptions made by our model? For instance, mention how all particles have same size.
- What are the initial conditions?
- PDE, Euler's method?
- Large number of particles → solved in reasonable time only using parallel computation.
- What do we want to evaluate? Scalability and performance?
- How is the paper organized?

1.2 OpenACC

- General description of it.
- What is it?
- Why are we using it?
- How are we using it?

2 Simulation methodology

2.1 Input files and preprocessing

- What are the arguments accepted by the program and why are they relevant?
- Pseudo-random initialization of particles.
- Explanation of particle interactions.
- Explanation of OpenACC implementation.

2.2 Postprocessing and visualization

- Explain how, for every frame, we keep the position $\mathbf{x} \in \mathbb{R}^3$ of all particles. Possibly mention something about the format of the VTK files.
- Process each of the files using ParaView.
- Possibly mention what we know about ParaView and what is relevant about it. Why does it help us?

3 Experimental results

3.1 Credibility of Particle Simulations

- Parallel vs serial results.
- Comparison against state-of-the-art software? (accuracy?)
- We could compare our results with a real system (e.g., Earth and Moon), but simplicity of our model may not allow for it (let's test such a case?).

4 Benchmarking and OpenACC Results

- How many GPUs were used? How many CPUs were used? What is the exact model used?
- How great was the speedup from serial to parallel?
- Efficiency...how do we measure it?

4.1 Roofline

- What is it?
- What information does it convey?
- How does it help us?

4.2 Benchmarks

- What is the complexity of the problem we explored?

4.3 OpenACC

- Having implemented OpenACC into the code, did we observe a speedup? How big was this speedup?

5 Conclusion

- Particle simulations: Why are they important? What problem(s) do they solve?
- GPU vs CPU performance; roofline model and benchmarks.
- Type of GPU and CPUs used?
- Thanks to...