

# Particle Simulations with OpenACC: Speedup and Scaling

## Status Update

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

(University of Lugano)

March 27, 2018

# Tasks so far

---

Our tasks for this week were the following:

- Develop serial code.
- Investigate visualization tools.
- Investigate parallelization methods.

# Serial code

---

The serial code is divided into the following three main sections:

- 1 Tracing particles (trails or no trails).
- 2 Drawing particles.
- 3 Updating particle details such as position and velocity.

For the moment, particle movement doesn't strictly abide to any well-established physics. For instance, particle collisions are not calculated yet.

# Serial code

---

Demo...

# Visualization tools

- For the time being, the simulation is done in two dimensions. This makes it relatively straightforward to paint the particles on the canvas.
- In three dimensions, we would need to add behaviour for the third dimension and would need to change the way in which particles are currently being drawn.
- To avoid building code for rendering in more than two dimensions, we could rely on visualization tools.

# Visualization tools

An option for visualization is using *ParaView* [2].

- Used at the CSCS (Swiss National Supercomputing Centre) [1].
- Open source, used for visualizing two and three-dimensional data sets.
- Platforms supported range from single-processor workstations to multiple-processor distributed-memory supercomputers or workstation clusters.
- Many file formats supported.

# Visualization tools

An example file format is the following:

<i>particle index</i>	<i>x-coordinate</i>	<i>y-coordinate</i>	<i>z-coordinate</i>
0	3.5	6.0	50.0
1	1.4	3.0	10.0
⋮	⋮	⋮	⋮
1000	5.5	10.5	20.2

# Visualization tools

Using the file format [particle index, x-coordinate, y-coordinate], we need to create a file for each time step. In each file, we'll write the position of each particle in the given time step.

- positions\_0.txt
- positions\_1.txt
- positions\_2.txt
- And so on...



# Visualization tools

---

Demo...

# Parallelization methods

Preliminary benchmarks made for matrix–matrix multiplication execution time, comparing performance of serial code vs code parallelized using *OpenACC*.

# Parallelization methods

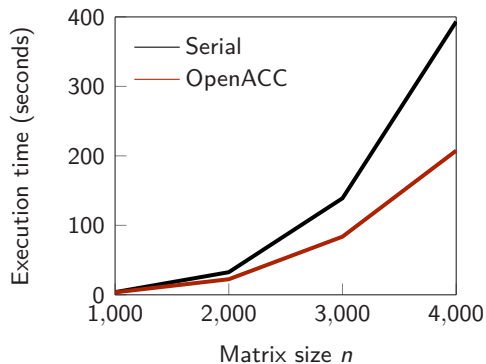


Figure: Matrix–matrix multiplication: execution time comparison.

# Project plan

---

Show current project calendar...

# References

---



**CSCS.** *ParaView*. URL: [https://user.cscs.ch/scientific\\_computing/supported\\_applications/paraview/](https://user.cscs.ch/scientific_computing/supported_applications/paraview/).



**ParaView.** *The ParaView Tutorial*. Nov. 2017. URL: [https://www.paraview.org/Wiki/The\\_ParaView\\_Tutorial](https://www.paraview.org/Wiki/The_ParaView_Tutorial).

# Questions

---

Thank you for your attention!  
Any questions?