Project Topic

for the Course

Parallel Algorithms II

February 9, 2021

1 Game Rules and Project Evaluation

All the points below must be met:

1. C++11 and above is accepted. C++17 is a recommended version (e.g. auto_ptr is removed, ...)
2. CUDA Toolkit 11 is a recommended version.
3. All project must be unique from the perspective of source code. The same or very similar solutions will be excluded from the evaluation and authors will be not classified.
4. The source codes must be written with respect to versatility, high performance, and minimal memory consumption; e.g. no wasted lines of codes, using generic classes (templates), using suitable data types/structures, avoiding memory leaks, using exceptions, etc. All these aspects will be evaluated.
5. Students have to provide sample data with relevant description. This data will be used in an application verification process.
6. The limitations of the application must be clearly defined, if any.
7. The solution must be consulted with the lecturer at least once during the semester.
8. All projects must be submitted in a form agreed with the lecturer until the end of organized education of the semester. The lecturer may change this deadline.

Code purity, application performance, and memory requirements will be included in the project evaluation.
The project can be considered as a competition. The better solution, the more points you can get in comparison with your colleagues.

2 Project Description

Imagine, that you have an area represented by its heightmap, e.g. in form of a grayscale image. This image is large enough but at least 16000x9000 pixels. Then, there is a large set of particles moving on this heighmap. All this particles are called followers looking for their leaders. Then, there is a set of leaders. Each leader is also a particle moving on the heightmap. Moreover, the following points describe movement of all particles:

- Moving of all particles respects basic physical laws that you can simplify, but some laws/rules must persists, e.g. a particle moving downhill increases its speed with respect to the slope of the terrain. The terrain slope must be calculated from the heightmap.
- Each follower follows the leader closest to him at the time. This can change very fast depending on the speed of particles.
- Each follower falling into a given leader's radius will be terminated.
- Particles never escape the area and never stops at its borders.
- There exists some speed factor that can limit movement of all particles.

It is important that your application can be run from the command line with one parameter. This parameter will be a JSON file with all the settings, and must have at least the following structure:

```
{
"leaders": 10,
"followers": 1000000,
"heightmap": "path_to_heightmap"
"heightmapGridX": 16000,
"heightmapGridY": 9000,
"leaderRadius": 50,
"speedFactor": 0.123,
"outputFile": "path_to_output_file"
}
```

It is evident that the project topic covers several aspects of parallel programming. All of them will be discussed. The output file should contain all your partial measurements, e.g. memory consumption, algorithm performance, etc.