

Project Milestone Report

Jiyu Hu (jiyuh)

Zhaohong Lyu (zhaohonl)

Schedule

4/3 - 4/7

Research for feasible approaches

4/10 - 4/14

Implement sequential implementation

4/17 - 4/21

Draft on parallel implementation

4/20 - 4/22

Optimize on parallelization - Jiyu

4/23 - 4/26

Benchmark and refine - Jiyu

4/27 - 4/30

Work on final report - Zhaohong

5/1 - 5/4

Design poster and showcase - Zhaohong

Status Update

We designed a basic framework for our project, which served as the foundation for all subsequent work. As part of our research, we explored several possible approaches for rendering fireworks, and evaluated their respective advantages and disadvantages.

Then we focused on the OpenGL display function. We leveraged the capabilities of CUDA by binding a surface object to an OpenGL texture, allowing us to reduce data movement and avoid copying image data from the renderer to OpenGL. This optimization enabled us to achieve improved performance and efficiency in our fireworks rendering.

We are almost finished with the sequential rendering routine. We plan to test the performance and rendering effect of the sequential program during this weekend. From there, we can know to what extent we need to parallelize the program.

Goals and Deliverables Alignment

Overall, we are making good progress with respect to the goals and deliverables. However, we have encountered some challenges along the way, particularly with regards to achieving high-quality rendering of our fireworks. As a result, we were a little behind our original schedule.

We might not be able to implement the reflection effect of the fireworks on a lake scene. However, we are still committed to achieving real-time rendering. So here are our adjusted goals.

Plan to achieve:

- Sequential version of the code that renders realistic fireworks as baseline
- Highly parallel implementation that achieves the same rendering effect with significant speed up
- Detailed analysis of the speedups and bottlenecks of the parallel program
- Showcase of firework simulation result during poster session

Nice to have:

- Real-time rendering of the firework simulation with high frame rates (60 - 120 fps)

Poster Session

We plan to showcase a demo of our fireworks renderer. We will also provide some diagrams that illustrate the workflow of our project, as well as some insights into the parallel optimization techniques that we have used to improve performance.

Preliminary Result

We didn't have any preliminary results at this point as we were focusing on refining the rendering effect of our fireworks renderer. However, we are optimistic about finishing the sequential version of the program during this weekend as we have figured out two promising methods to calculate the rendering object via CUDA and display the objects in real-time through OpenGL. We hope to implement frame rate calculation for benchmark soon.

Concern

We are unsure if we will be able to render the fireworks in real-time, as this can be a computationally intensive task that may require significant optimization efforts.

Additionally, we are concerned about the rendering effect of our fireworks, particularly with regard to shading and lighting effects. There are still blocks and uncertainties for us to achieve the desired visual effect.