Midterm: AlienGLRenderer: Cloth Simulation

Arrian Chi

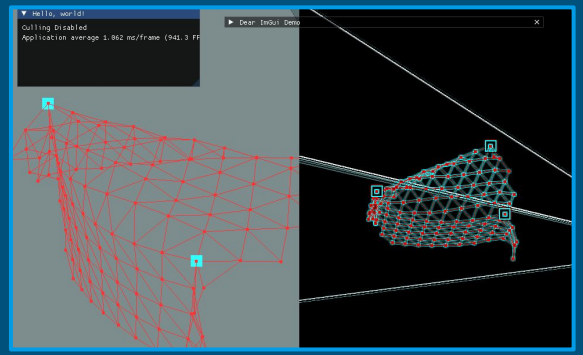
Link: https://docs.google.com/presentation/d/1VFmNMyhTEvu0-XLXWqgcFGkal4t_rovTlGv1VSKVTzs

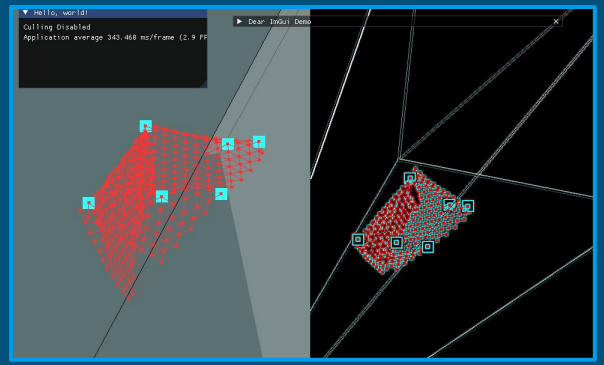
Insert video of simulation here

I havent recorded a good one…

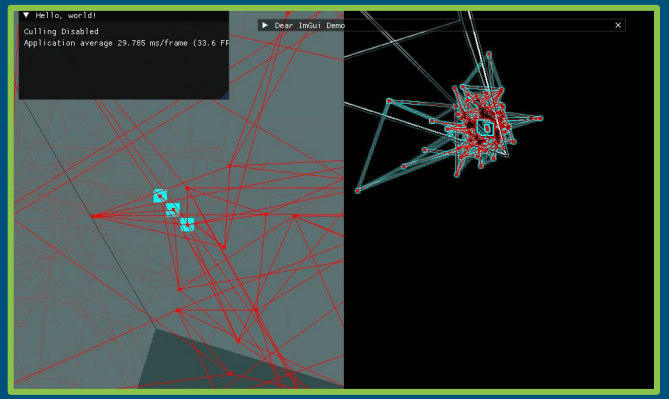
Progress

- Created base particle system cloth is based on
- Created a triangular mesh to discretize cloth
- Implemented the simulation loop (single-threaded)
  - Runs in less than more than 30 FPS for simulations with less than 121 nodes
  - Simulates fairly well in release build until 361 nodes (more than 1000 DOFs)
- This is without any sort of multithreading involved!!!





Comments



- Progress was a bit slow
  - My renderer's design is brittle...
    - Caught up in C++ templates and system design
  - OpenGL sucks
    - Memory alignment issues with vec3 and the DEP algorithm's use of vec3
  - My renderer and Python/Numpy are very different
    - Numpy uses double (15 digits of precision)
    - My renderer uses float (7 digits of precision)
  - Copying an algorithm over to another is prone to error
    - Hours of debugging and screaming at a screen
  - Some degeneracies occur...

Revised Plan

Week 4: Optimize the simulation algorithm

- With memoization (reducing redundant calculations)
- Multi-threading
  - Force/Jacobian calculation (Turn Eigen's multithreading option on)
  - Using the PARDISO solver (Compare Eigen's and Panau)

Week 5: Present findings in class

Week 5+: Capstone?

References

1] D. Baraff and A. Witkin, Large Steps in Cloth Simulation. New York, NY, USA: Association for Computing Machinery, 1 ed., 1998.

[2] E. Grinspun, A. N. Hirani, M. Desbrun, and P. Schr¨oder, “Discrete shells,” in Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, SCA ’03, (Goslar, DEU), p. 62–67, Eurographics Association, 2003.

[3] R. Tamstorf and E. Grinspun, “Discrete bending forces and their jacobians,” Graph. Models, vol. 75, p. 362–370, Nov. 2013.

[4] O. Schenk, K. G¨artner, W. Fichtner, and A. Stricker, “Pardiso: a high-performance serial and parallel sparse linear solver in semiconductor device simulation,” Future Generation Computer Systems, vol. 18, no. 1, pp. 69–78, 2001. I. High Performance Numerical Methods and Applications. II. Performance Data Mining: Automated Diagnosis, Adaption, and Optimization. 2001.

[5] J. Bolz, I. Farmer, E. Grinspun, and P. Schr¨oder, “Sparse matrix solvers on the gpu: conjugate gradients and multigrid,” ACM Trans. Graph., vol. 22, p. 917–924, July 2003.