Parallel Spectral Renderer
Stuart Maier
Computer Systems Lab 2009-2010

Abstract

Computer generation of highly realistic images has been a difficult problem. Although there are algorithms that can generate images that look essentially real, they take large amounts of time to render. This project explores ways of distributing that onto multiple computers, in order to speed up the process.

Background and Introduction

This project is a combination of two parts: the graphics part and the parallel part. The graphics part is based off of established work in computer graphics rendering. The main method of generating realistic computer generated images is the path tracer, which shoots off rays to generate the scene. I am also using spectral rendering, which replaces the standard RGB model with actual wavelengths of light. The parallel part is based off of the BOINC framework, which works with independent clients which only talk to the server at the beginning and end.

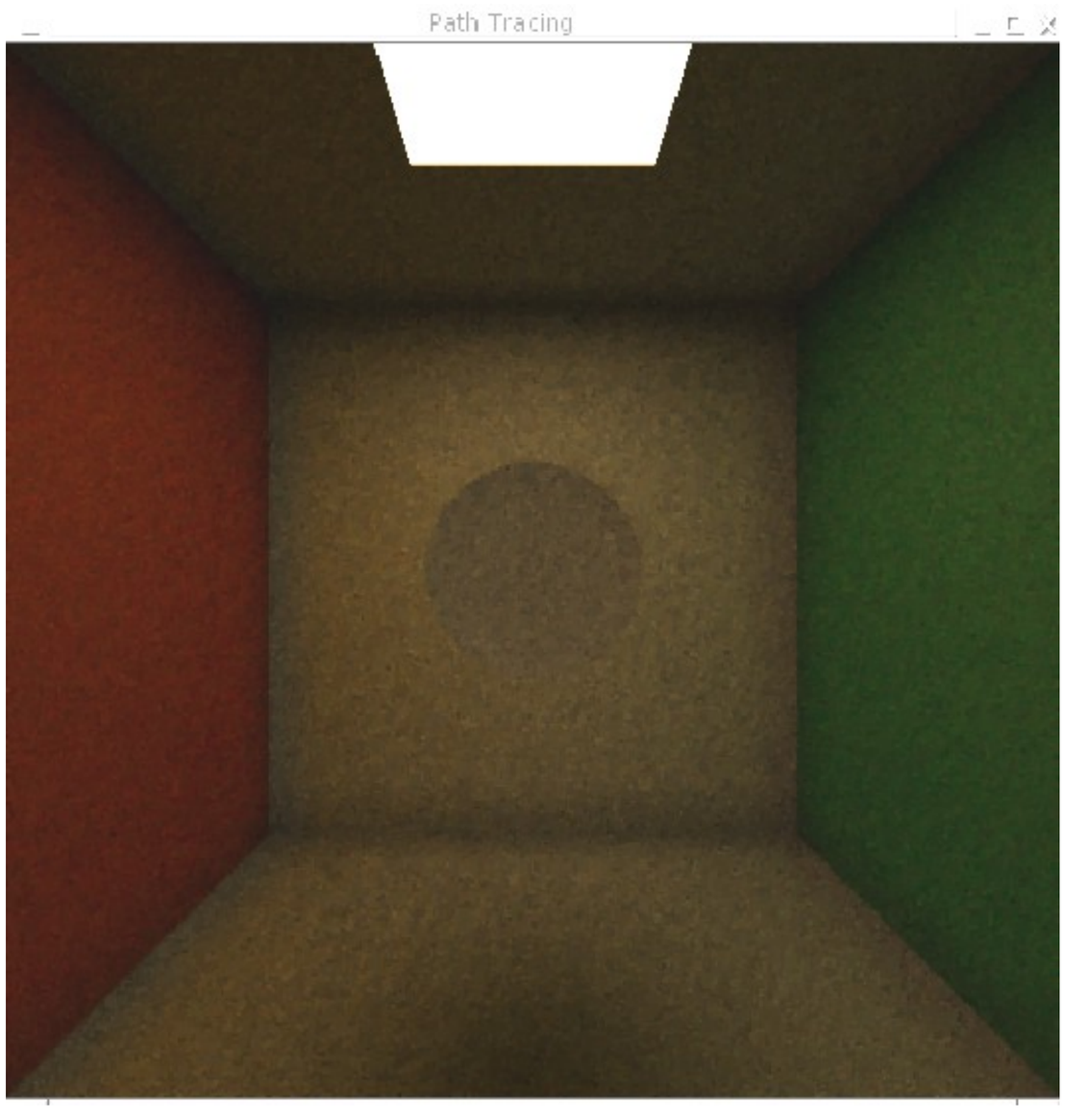

Fig 1: The final output of a rendered scene.

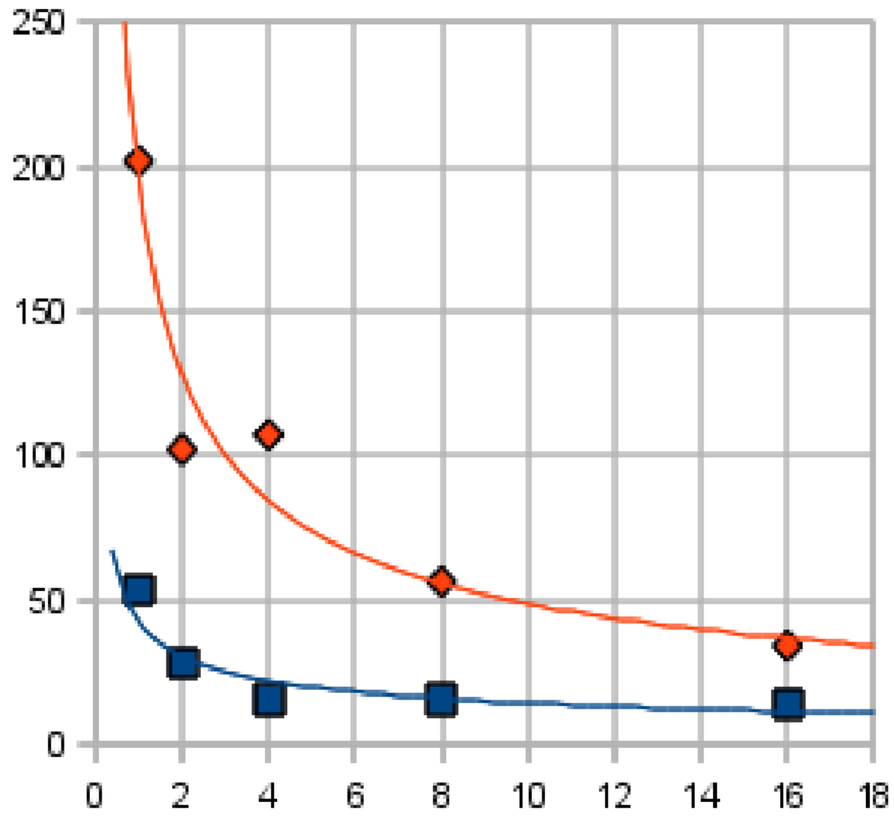

Fig 2: The speedup gained by using parallelism.

Discussion

Currently, the path tracing part of the project has been implemented, as seen above. It is able to generate a scene that looks realistic and it exhibits properties of real light. It is also rendering these colors spectrally, using 81 different wavelengths in order to model light.. As far as the parallelism goes, there is a somewhat developed system that provides adequate parallelization as well as speedup, but there is room for improvement by changing to a more robust parallel system.

Results and Conclusions

Currently, I only have parallelism results from the experiment. I have found that using multiple clients does cause speedup. However, there are bottlenecks. One is the delays caused by network lag. Another is that some clients have more work than others, so some clients are finished and idle while other clients are still processing their part of the image.