In figures like the Mandelbrot set (right) and numerical integration (below), different threads can show their contribution to the computation.

In a figure showing four Langton’s ants, each ant’s trail is drawn by a different thread:

TSGL: A Thread-Safe Graphics Library for Creating Multithreaded Visualizations
Joel Adams, Patrick Crain, Mark Vander Stel, Dept of Computer Science, Calvin College
https://github.com/Calvin-CS/TSGL

Abstract

Thanks to the ACM/IEEE CS Curriculum 2013 recommendations, parallel computing is now in the undergraduate core computer science (CS) curriculum. CS educators face a new challenge: teaching parallel computing concepts to undergraduates. One way to meet this challenge is to create parallel programs that use real-time graphics to help students visualize the program’s underlying parallel behavior.

TSGL is a free, platform-independent C++ graphics library designed to make it easier for anyone to create visual representations of multithreaded behavior. The library provides 2D Canvas classes within which different threads can safely manipulate different parts of a digital image, and/or draw shapes using different colors. The library can thus be used to extend a multithreaded application with a visualization that shows what each thread is contributing to the problem’s solution. The library includes several examples (e.g., image-processing, the Mandelbrot set, etc.) that illustrate its use.

Methods

Dependencies:
TSGL was built on top of OpenGL using GLFW [1] as the window manager. It uses to libpng [2] and libjpeg [3] to load images.

TSGL UML Diagram:

Interfaces:
• A Canvas class provides a window to draw on.
• A CartesianCanvas class (a subclass of Canvas) simplifies the task of plotting functions on a Cartesian coordinate system.
• A Shape class hierarchy includes subclasses for the various shapes that can be drawn.
• A Function class hierarchy includes subclasses for commonly-used functions.
• A Timer class facilitates delaying a thread for a specified interval, to slow the rendering sufficiently to visualize the actions of individual threads.
• A Color class facilitates giving each thread a distinct color for drawing.

Examples

In figures like the Mandelbrot set (right) and numerical integration (below), different threads can show their contribution to the computation.

Cosine integration with eight threads

Above: Cosine integration with 8 threads partway through their computations, each thread coloring its piece of the area with a different color in real-time.
Below: The same figure, all threads finished.

Completed

Langton’s ants with four threads

In a figure showing four Langton’s ants, each ant’s trail is drawn by a different thread.

References


This work made possible by NSF DUE #1225739.