

TSGL: A Thread Safe Graphics Library for Visualizing Parallelism

Joel C. Adams

Patrick A. Crain

Mark B. Vander Stel

Department of Computer Science

Calvin College

Parallel Multiprocessors



The Challenge

How can we help students understand parallel computing concepts, so that they can design and implement scalable parallel programs, in order to more quickly solve today's science problems?

Conjecture: Many students are visual learners; if we can help them *visualize parallel behavior*, that will help them understand and master parallelism more readily.

TSGL

... is a *thread-safe graphics library*:

- Object-oriented; written in C++11
- Threads can safely draw on the same *Canvas*, which is updated in approximate real-time
- Allows instructors (and students) to create *visualizations showing exactly what each thread is contributing to a parallel computation*

The Basic Idea

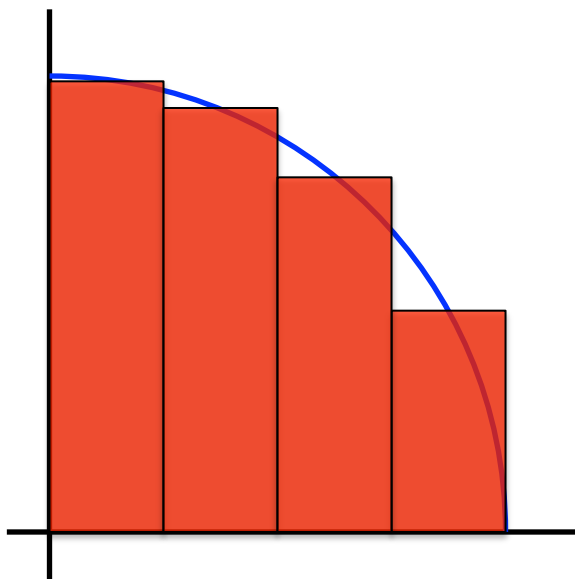
Given a working program (multithreaded or otherwise):

- Add a *Canvas* to the program
- Have each thread draw on the *Canvas* in a way that visually illustrates its behavior
- Run the program to view the threads' behaviors.

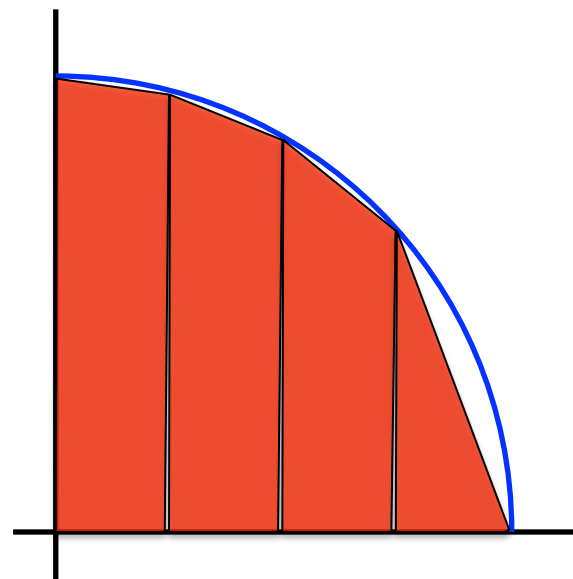
Common Teaching Example

Approximate π as 4 x the integral of the unit circle (aka “the area under the curve”)

Rectangle Approach:



Trapezoid Approach:



More intervals \rightarrow more accuracy, but longer to compute

Annotating the (Pseudo)Code

```
double Integral::useRectangleMethod() {  
    area = 0.0;  
  
    #pragma omp parallel  
    {  
        Canvas canvas(...);  
        Color threadColor = getColor(threadID);  
        #pragma omp for  
        for ( x = a; x < b; x+= width) {  
            y = f(x);  
            area += (y * deltaX);  
            canvas.drawRectangle(x, 0, width, y, color);  
        }  
    }  
    return area;  
}
```

Demo Video



TSGL Features

- Object-Oriented:
 - *Canvas* and *CartesianCanvas* classes support drawing *Shape*, *Function*, *Image*, ... objects
- *Timer* class provides precise control to slow thread execution, when desired.
- Works with OpenMP, POSIX, C++11, ... threads
- HTML-based API documentation (Doxygen).
- Currently workable, but still evolving!

Conclusion

- TSGL is a tool designed to help students/faculty *visualize multithreaded behavior*, and learn to write their own parallel programs.

Resources

- TSGL may be freely downloaded from <https://github.com/Calvin-CS/TSGL>.
- Thanks to:
 - The U.S. National Science Foundation
 - You!

