University of the Witwatersrand

School of Computer Science and Applied Mathematics

COMS3008: Parallel Computing Lab Assignment 3

12 August 2016

By Chalom, J. (711985)

1 General Assumptions and Methodology

All drawn diagrams were drawn using `http://draw.io/`, and all results from empirical analysis, were statistically analysed by the data processing program used in lab 1. Those averaged results were then plotted using Microsoft Excel.

The system used for empirical analysis, had an i3 Ivy-bridge CPU, with four processing units and 8GB of RAM. This system can open up a maximum of 26000 threads of computation for a given program excluding any background and operating system tasks.

1.1 Glossary Of Terms:

Trapezoid - is defined as the British definition (Trapezium) as a quadrilateral with no sides parallel
IBT - is an acronym for integration by parts.
STR - is an acronym for Simpson's Trapezoidal Rule.
CTR - is an acronym for Composite Trapezoidal Rule.

2 Analytical Solution of Given Integral

Let $f(x) = \int_0^{20} xe^{-x}dx$

By IBT let $\int fdg = fg - \int gdf$,
where $f = x$, $dg = e^{-x}dx$, $df = dx$ and $g = -e^{-x}$.

$$\therefore \int xe^{-x}dx = -e^{-x} + \int e^{-x}dx \tag{1}$$

Let $u = -x$ and $dy = -dx$

$$\begin{aligned} \therefore -e^{-x} + \int e^{-x}dx &= -e^{-x}x - \int e^{u}du \\ &= -e^{u} - e^{-x}x + c \\ &= -e^{-x} - e^{-x} + c \end{aligned} \tag{2}$$

$$\begin{aligned}
\therefore \int_0^{20} xe^{-x}dx &= \left[-e^{-x} - e^{-x} + c\right]_0^{20} \\
&= -e^{-20}(20) - e^{-20} + e^0(0) + e^0 \\
&= 1 - e^{-20}(20) - e^{-20} \\
&= 1 - \epsilon \\
&\simeq 1,00
\end{aligned} \tag{3}$$

Note: $\epsilon \simeq 4,329422607x10^{-8}$.

3 Parallel Trapezoidal Integration

3.1 The Trapezium:

Let a be the shortest side of a trapezoid (Fig 1), b be the longest side and h be the base of the trapezoid (Fig 1).

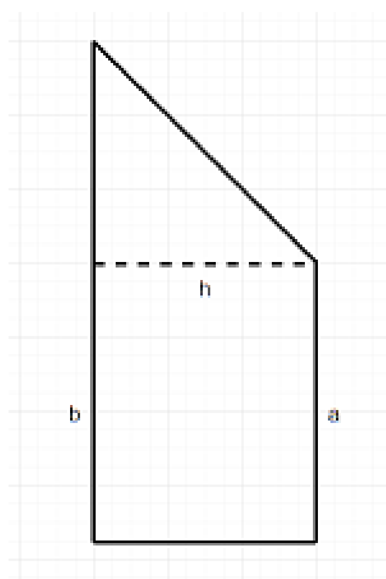


Fig 1. An example trapezium

3.2 The area of a trapezoid is defined as:

$$
\begin{aligned}
A &= (a.h) + (\frac{1}{2}(b-a).h) \\
&= h.(a + \frac{1}{2}(b-a)) \\
&= h.(a + \frac{1}{2}b - \frac{1}{2}a) \\
&= h.(\frac{1}{2}a + \frac{1}{2}b) \\
&= \frac{a+b}{2}h
\end{aligned} \tag{4}
$$

3.3 Derivation of the rule:

Let $f(x)$ be a function where $x \in [a, b]$, such that
$a = x_0 < x_1 < ... < x_n = b \in \mathbb{Z}$

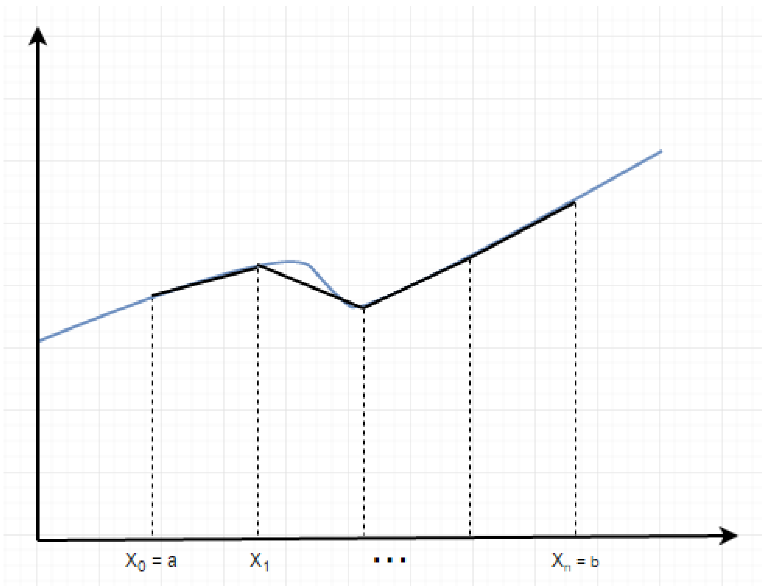


Fig 2. Approximation of a polynomial

Note:

$$
\begin{aligned}
h &= x_{n+1} - x_n \\
&= \Delta x
\end{aligned} \tag{5}
$$

The area of the n^{th} trapezoid is given by:

$$A_n = \Delta x \left[\frac{f(x_i) + f(x_{i+1})}{2} \right] \tag{6}$$

Since Δx is fixed for each partition as it is assumed that there are a uniform number of trapezoids under $f(x)$,

$$\begin{aligned} \Delta x &= \frac{b-a}{n} \\ \therefore A_n &= \left[\frac{f(x_i) + f(x_{i+1})}{2} \right] \end{aligned} \tag{7}$$

Let $m = n - 1$, This implies that

$$\begin{aligned} \int_a^b f(x)dx &\simeq \sum_{i=0}^{m} A_i \\ &\simeq \sum_{i=0}^{m} \Delta x \left[\frac{f(x_i) + f(x_{i+1})}{2} \right] \\ &\simeq \Delta x \sum_{i=0}^{m} \left[\frac{f(x_i) + f(x_{i+1})}{2} \right] \\ &\simeq \frac{b-a}{2n} \sum_{i=0}^{m} \left[f(x_i) + f(x_{i+1}) \right] \end{aligned} \tag{8}$$

3.4 Method of Implementation:

The derivation of the Trapezoidal rule (from equation 8) was used and both serial and parallel implementations were programmed. The serial function's results are the baseline for the analysis. The parallel decomposition was a standard openMP parallel for directive where a reduction on the approximation was specified.

The error calculated in these experiments is the result of subtracting the analytical result derived in section 2 by the final approximation at each iteration. If the error is negative then the specific step approximated the integral under the analytical result otherwise it was over the result.

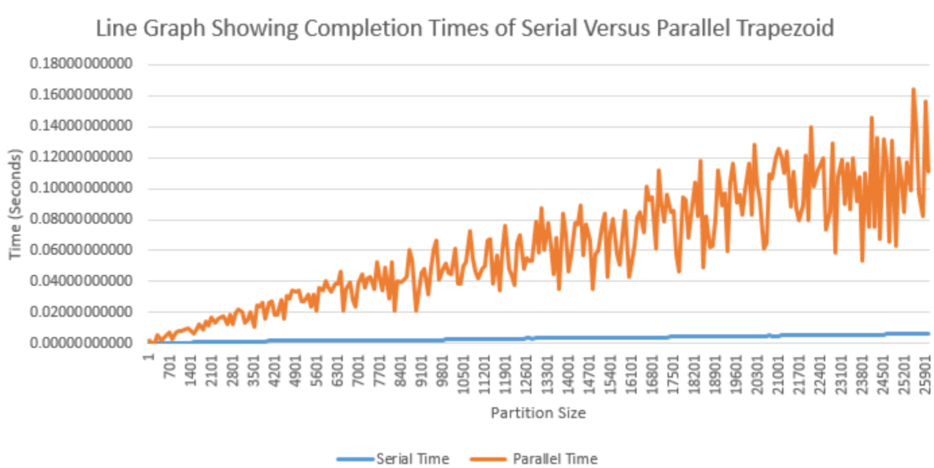
Three result sets were captured. One was the serial implementation where only the main thread was used. This algorithm starts at 1 partition and increments by 100 for 260 iterations. These are the result data points generated.

In the two parallel result sets, one looked at a fixed number of threads (4 threads) and an incrementing number of partitions and the other looked at a fixed number of partitions (1000000) and an incrementing the number of threads from 1 to 26000 threads, at 100 increments 260 times.

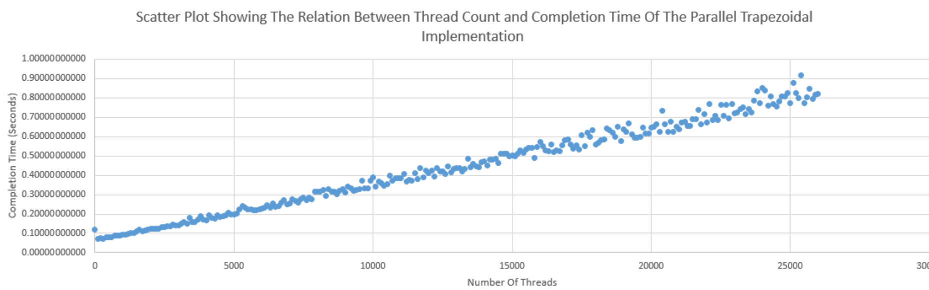
All results included the time each specific trapezoidal approximation took to complete, either the number of threads or the number of partitions, the error, and the absolute error.

3.5 Implementation Results:

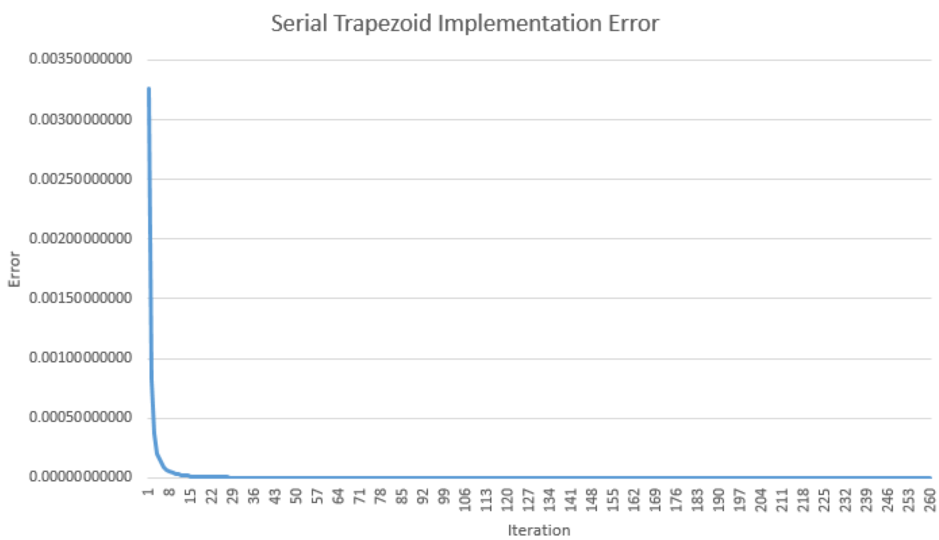
The results of this empirical experiment show that this parallel decomposition of the Trapezoidal Rule is not as efficient or fast as the serial implementation used.



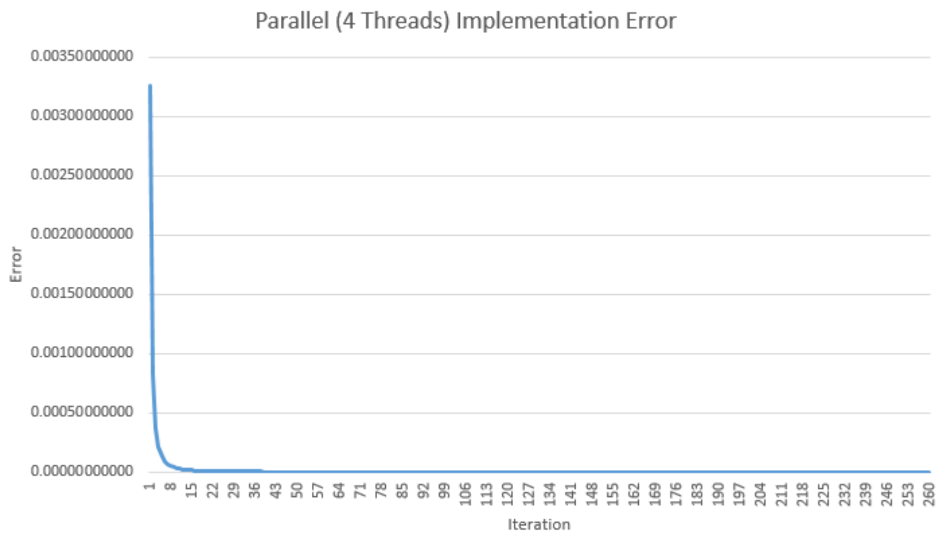
Another result shows that as the number of threads of the parallel implementation increase so does the completion time of the rule (At a liner rate).



The error of both the parallel implementation of the trapezoid and its serial counterpart have very similar exponentially decreasing errors.



Trapezoidal Serial Error



Trapezoidal Parallel Error

3.6 Implementation Analysis:

The results produced show that a different parallel decomposition would be required as the one tested here are far less efficient or desirable compared to the results produced by the serial implementation. There is also a problem related to iso-efficiency where there is a linear increase in the time complexity of the parallel implementation as the number of threads increases. The respective errors of both the serial and the parallel are comparable and have exponentially decreasing error as the number of partitions increases.

4 Parallel Monte Carlo Integration

4.1 Method of Implementation:

A boundary was setup which went from $a = 0$ to $b = 20$ on the X axis and from c to d on the Y axis. c, d are the results of $c = f(a) + 1$ and $d = f(b) + 1$. This produces a rectangle where part of its area is the integral of the function defined in section 2. Random x and y values were generated for every particle in the experiment. They were generated in the range $[a, b]$ and $[c, d]$ respectively. The experiment was conducted using a purely serial function and also a parallel function. The serial function's results are the baseline for the analysis. Each increment taken was repeated 25 times and after the experiment was run averages and other statistics were taken and those results were used to plot the graphs in the results subsection.

The error calculated in these experiments is the result of subtracting the analytical result derived in section 2 by the final approximation at each iteration. If the error is negative then the specific step approximated the integral under the analytical result otherwise it was over the result.

The serial implementation iterated through the number of particles per experiment (from 1 to 26000, at 100 increments) and returned the approximation, an error, the time taken by each calculation and an absolute error.

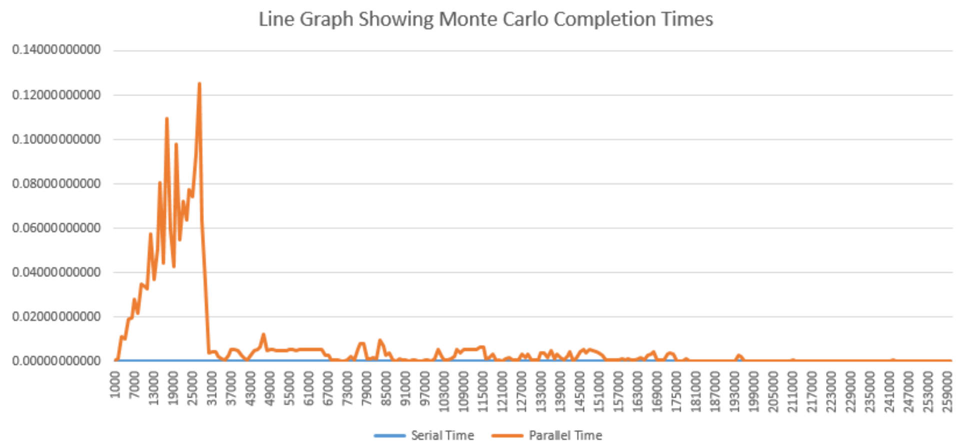
Both Parallel implementations made used of the same decomposition using openMP's parallel for directive. The private variable for each execution unit was the result of the randomly generated point as the points were never stored in memory for later use. There were shared variables where were c and d. A reduction operation was also done on the only variable used outside of the parallel code block, which was variable counting the number of points which occurred below the graph (in side the area of the graph), as the total number of points was known before-hand.

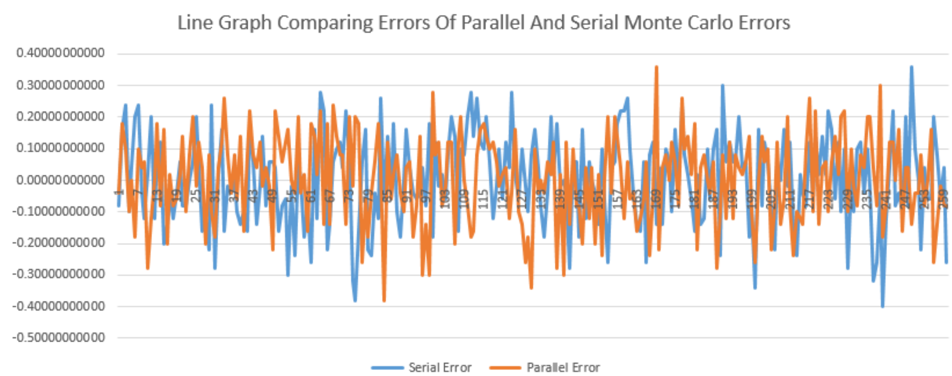
The one parallel experiment locked the number of threads (4 threads) and incremented the number of particles which to generate from 1 particle to 26000 particles incrementing by 100 each time. The other one locked the number of particles at 1000000 and incremented the number of threads from 1 thread to 26000 threads, incrementing by 100 at each step.

4.2 Implementation Results:

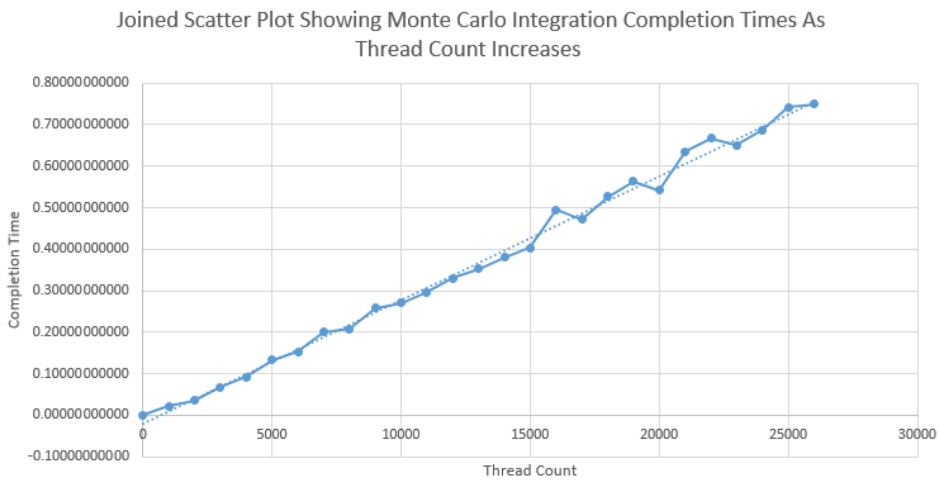
The results of this empirical analysis show that initially the parallel implementation of the Monte Carlo Integration algorithm is very unpredictable versus the serial version. As the number of partitions increase the parallel implementation's time complexity tends towards imitating the serial one.

The error trends of both the serial and parallel implementations are comparable. The almost random nature and oscillation between positive and negative errors is noted. This shows that there is both under and over estimation occurring at different increments.





Again it can be seen that as the thread count increases so does the time complexity of the parallel algorithm.



4.3 Implementation Analysis:

Again the results show that the decomposition used in this analysis did not have desired results. The serial equivalent was by far more reliable and faster. These results can also be due to system interference or problems in the respective implementations. The reliability of this parallel decomposition is also in question. It can be seen from the completion times graph that this implementation is produces unstable results. Again iso-efficiency can be observed where the increase of threads produces a linear trend.

5 Conclusion

These experiments show that the current parallel decompositions are not effective in improving performance of these approximations of a finite integral. Better parallel decompositions and implementations must be used. Some experimental methods should also be evaluated and changed as necessary to provide more accurate results.

6 Appendix A: Alternative Derivation of Trapezoidal Rule

6.1 Derivation of the Trapezoidal Rule (STR):

Let P_n be a polynomial function of n terms.

$$\begin{aligned}\int_a^b P_n(x)dx &\simeq \int_a^b f[a]dx + \int_a^b f[a,b](x-a)dx \\ &\simeq (b-a)f(x) + (\frac{f(b)-f(a)}{(b-a)}).\frac{1}{2}(x-a)^2]_a^b \\ &\simeq (b-a)f(a) + (\frac{f(b)-f(a)}{(b-a)}).\frac{1}{2}(b-a)^2 \\ &\simeq \frac{b-a}{2}(f(a)+f(b))\end{aligned} \tag{9}$$

$$Error = \frac{-(b-a)^3}{12}f^{''}(\phi) \tag{10}$$

6.2 Derivation of the Composite Trapezoidal Rule (CTR):

For sub-intervals: $a = x_0 < x_1 < ... < x_n = b$

$$\begin{aligned}n &= \frac{b-a}{m} \\ &= \Delta x \\ &= (x_1 - x_0)\end{aligned} \tag{11}$$

Note: Δx is fixed.

$$\begin{aligned}\int_a^b f(x)dx &= \int_{a=x_0}^{x_1} f(x) + \int_{x_1}^{x_2} f(x) + ... + \int_{x_{n-1}}^{x_n=b} f(x) \\ &\simeq (\frac{x_1-x_0}{2})[f(x_0)+f(x_1)] + ... + (\frac{x_m - x_{m-1}}{2})[f(x_{m-1})+f(x_m)] \\ &\simeq \frac{n}{2}[f_0 + 2f_1 + 2f_2 + ... + 2f_{m-1} + f_m]\end{aligned} \tag{12}$$

$$Error_{CTR} = \frac{(b-a)^3}{12m^2}\ f^0(\phi),\ \phi \in [a,b] \tag{13}$$