

## Set 3 - OpenMP I

Issued: March 22, 2023

### Question 1: Parallel Numerical Integration

The sequential code in `mc_serial.c` computes the integral of a function using the Monte Carlo method. Implement in `mc_parallel.c` a parallel version of the code using OpenMP. Examine the scaling of your code.

Hints:

- Replace `drand48` with the thread safe `erand48` function <sup>1</sup>.
- Use a private buffer for `erand48` within each OpenMP thread.
- Use a different seed to initialize each buffer for `erand48`.

---

```
1 // prototype
2 double erand48(unsigned short xsubi[3]);
3
4 // declaration and initialization of buffer
5 unsigned short buffer[3];
6 buffer[0] = 0;
7 buffer[1] = 0;
8 buffer[2] = <thread related number>;
9
10 // usage
11 xi = erand48(buffer);
```

---

<sup>1</sup><https://pubs.opengroup.org/onlinepubs/007908799/xsh/drand48.html>

A possible OpenMP-based parallelization of the code can be found in `mc_parallel.c`. The strong scaling plot is depicted in Figure 1.

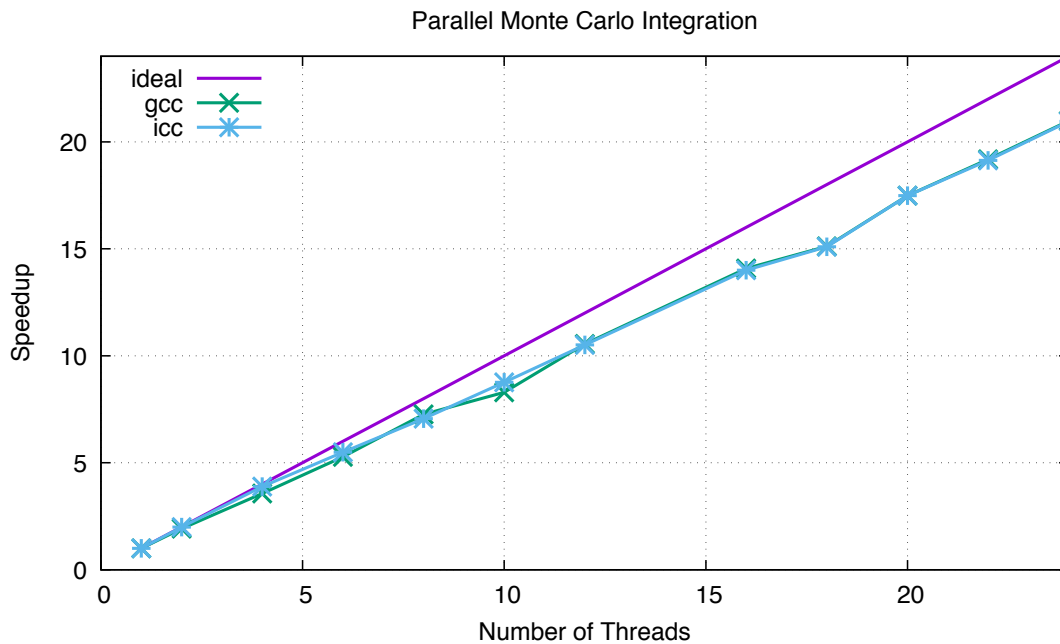


Figure 1: Strong scaling plot for the Monte Carlo integrator; Compiler optimizations are always -O3.