MPI Implementation

The Mandelbrot set was computed using the **MPI** parallelization technique with varying numbers of threads. The algorithm was ran in a similar manner to the **OpenMP** version (10 times per iteration value, same number of iterations and processes counts).

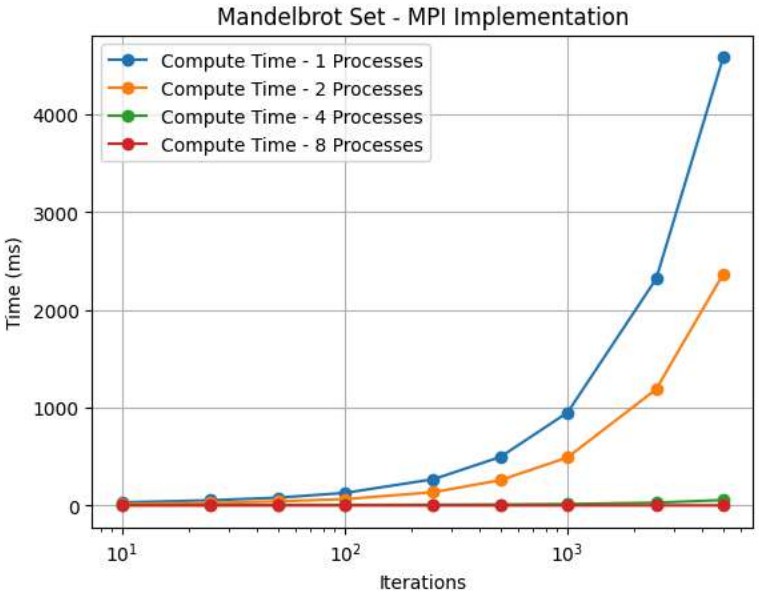
```
iteration_values = [10, 25, 50, 100, 250, 500, 1000, 2500, 5000]
```

The computation was run for a total of 10 times per iteration value, and the graph represents the **average** time taken per iteration for different thread counts.

The parallel version of the algorithm was tested with the following numbers of threads:

```
num_processes = [1, 2, 4, 8]
```

Performance Metrics



Data (*5700x3D*):

Threads	Iterations	Compute Time (ms)	Save Time (ms)
1	10	31.90	17.20
1	25	53.50	18.90
1	50	79.90	20.70
1	100	129.00	19.30
1	250	268.90	19.30
1	500	496.90	19.20
1	1000	950.60	19.60
1	2500	2316.50	19.20
1	5000	4580.00	19.20
2	10	15.90	17.00
2	25	26.50	18.30
2	50	42.50	19.00
2	100	65.40	19.20

Threads	Iterations	Compute Time (ms)	Save Time (ms)
2	250	136.10	19.30
2	500	259.70	19.30
2	1000	491.60	19.40
2	2500	1188.20	19.00
2	5000	2364.50	19.30
4	10	4.30	19.00
4	25	5.00	21.30
4	50	5.70	21.10
4	100	6.00	22.20
4	250	8.20	21.70
4	500	10.30	21.60
4	1000	15.20	22.50
4	2500	29.00	22.80
4	5000	56.10	22.30
8	10	1.30	20.80
8	25	1.00	22.30
8	50	1.40	21.70
8	100	1.20	22.70
8	250	1.40	22.90
8	500	1.50	23.60
8	1000	1.30	22.50
8	2500	1.50	22.80
8	5000	1.40	24.30