Haniye Kashgarani

Assignment 2

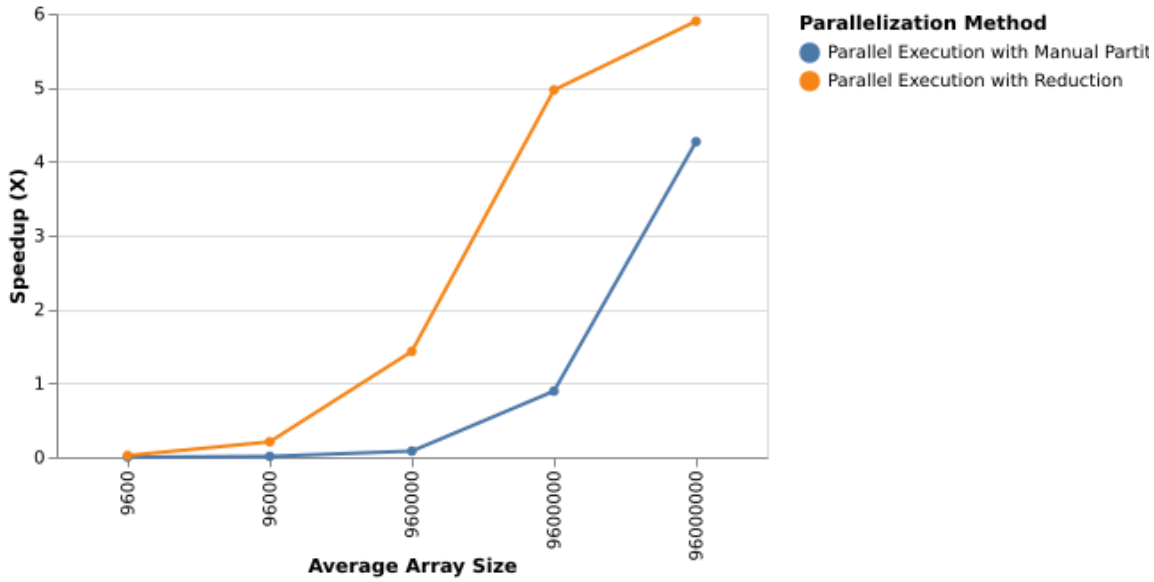
1) The number of the logical processors on the system is 12 and the number of threads spawned is 12 and equal to the number of logical processors on the system. The table below shows the timing statistics. By increasing the array size, the sequential time is increasing significantly, and in the first two cases, when the array size is smaller, the parallel execution with reduction runtime is worse than sequential time. But by increasing the array size the parallel executions with reduction have speedup comparing to the sequential runs. Also, parallel execution with manual partitioning is just beneficial when the array size is 96000000, and even when the array size is this much big, the parallel execution with reduction method is faster. So if the array size is small we should execute it sequentially and if it' large parallel execution with reduction method is more optimum terms of time.

Array Size	9600	96000	960000	9600000	96000000
Sequential Execution time (msecs)	0.01077	0.10969	1.13956	13.8471	126.454
Parallel Execution with Manual Partition time (msecs)	7.56528	9.1715	13.6246	15.4683	29.6244
Parallel Execution with Reduction time (msecs)	0.44657	0.52448	0.79515	2.78691	21.4321

2) The speedup achieved between the sequential and parallel versions using the average computation time.
Speedup = ts/tp

Array Size	9600	96000	960000	9600000	96000000
Parallel Execution with Manual Partition speedup	0.00142361	0.01196	0.08364	0.89519	4.26857
Parallel Execution with Reduction time speedup	0.02412	0.20914	1.4331	4.9686	5.9002

3) The plot of the variation of the speedup with increasing vector sizes for the 12 number of threads spawned



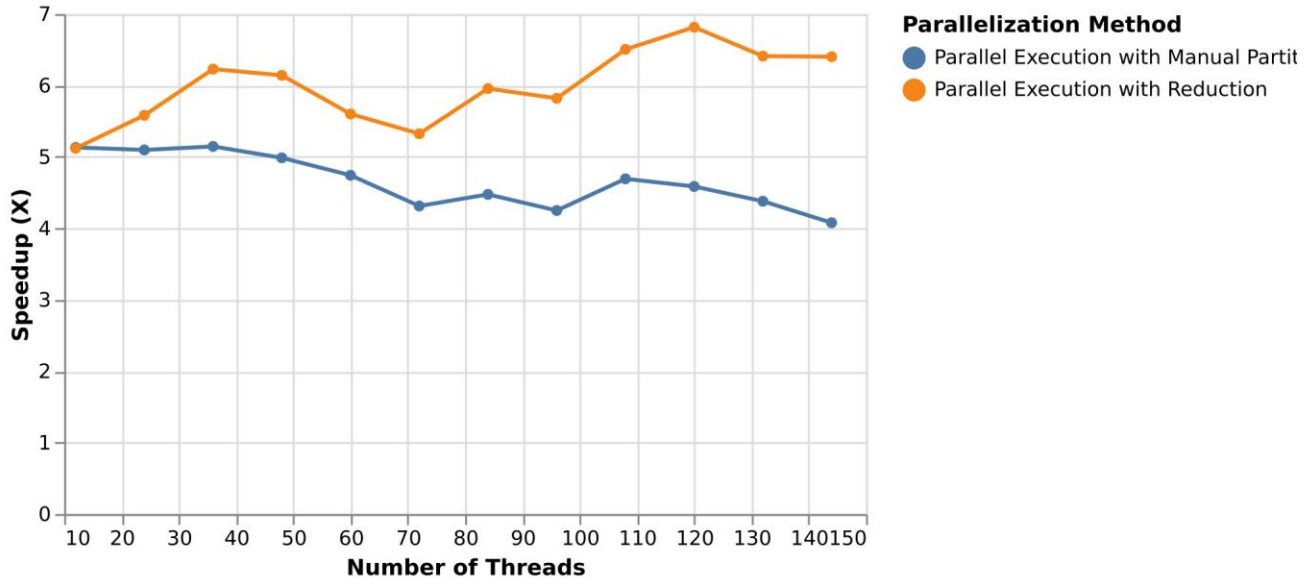
4) Runtime for a fix array size (96000000) and different number of threads.

Number of threads	12	24	48	72	96	120	144
Sequential Execution time (msecs)	134.613	136.993	136.86	122.279	128.543	147.739	133.108
Parallel Execution with Manual Partition time (msecs)	26.2303	26.8781	27.4488	28.3591	30.2526	32.2194	32.6492
Parallel Execution with Reduction time (msecs)	26.2905	24.5434	22.2845	22.9667	22.0879	21.6771	20.792

The speedup achieved between the sequential and parallel versions using the average computation time.

Number of threads	12	24	48	72	96	120	144
Parallel Execution with Manual Partition speedup	5.132	5.097	4.986	4.312	4.249	4.585	4.077
Parallel Execution with Reduction speedup	5.120	5.582	6.141	5.324	5.820	6.815	6.402

5) The plot of the variation of the speedup with fixed array size (96000000) and increasing number of threads.



By increasing the number of threads the speedup for parallel execution with manual partitioning is decreasing, but the speedup in parallel execution with reduction is increasing. This means whenever we want to have a large amount of threads, the optimal way is to use parallel execution with reduction.