2d) Table 1 below shows the computed volumes and standard deviation for different number of threads for 5e7 histories.

Table 1

Threads	Sphere Volume	Std. Deviation	Shell Volume	Std. Deviation
1	4189.1	0.565053	0.909193	0.000299192
2	4188.87	0.565054	0.909134	0.000299184
4	4189.37	0.565051	0.909	0.000299166

The number of processors is varied using pbs script, such that single thread runs on single processor.

It can be observed that, the round of error reduces as the number of threads increases. Each processor runs a fewer number of operations with increasing number of threads. This has lead to drop in standard deviation. Due to this, volumes calculated using simulation moves closer to the theoretically calculated volumes.

2e) Figure 1 shows the variation in the wall clock time with number of threads. With increasing number of threads the wall clock time increases.

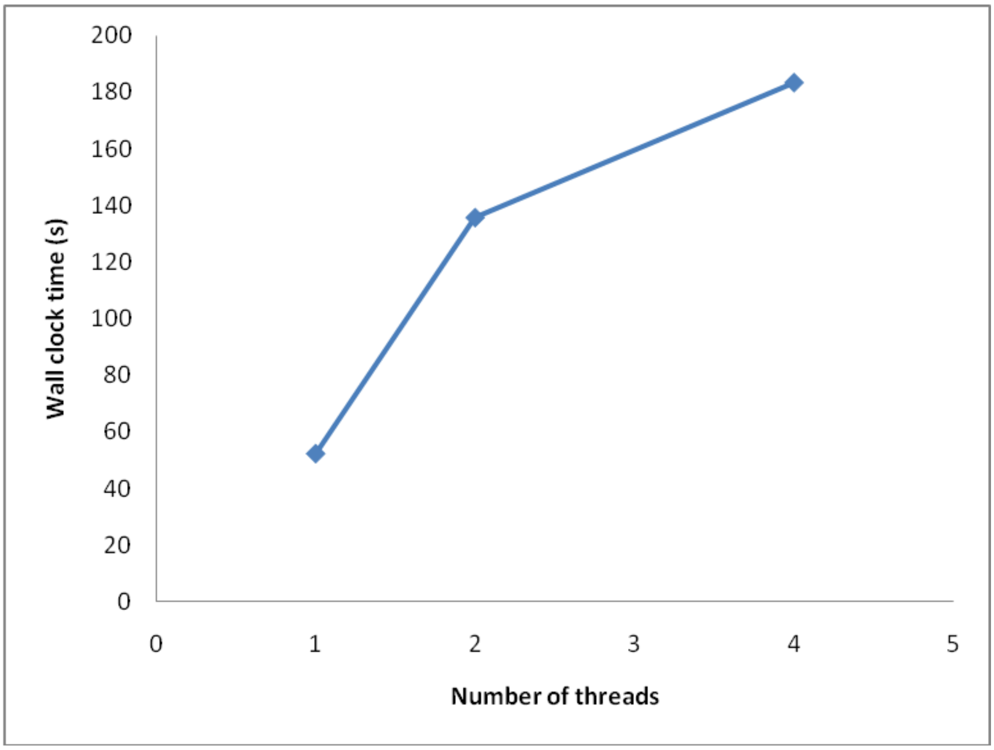

Fig. 1- Variation of number of threads with wall clock time

Wall clock time increases due to increase in overheads (communication, assemble data etc.) with increasing number of threads. Also there is large overhead of calling system function used for random number generation (srand()).

3b) For speedup calculation sequential code was used to calculate the time required for each history. As shown in Fig. 2 below, for all of the cases the speedup first drops rapidly from its initial value close to unity. This drop is due to increased overhead of parallelization. With further increase in number of threads, speed up increases, due to increase in gain from increased number of threads over overheads of parallelization. This gain diminishes further with increased number of threads.

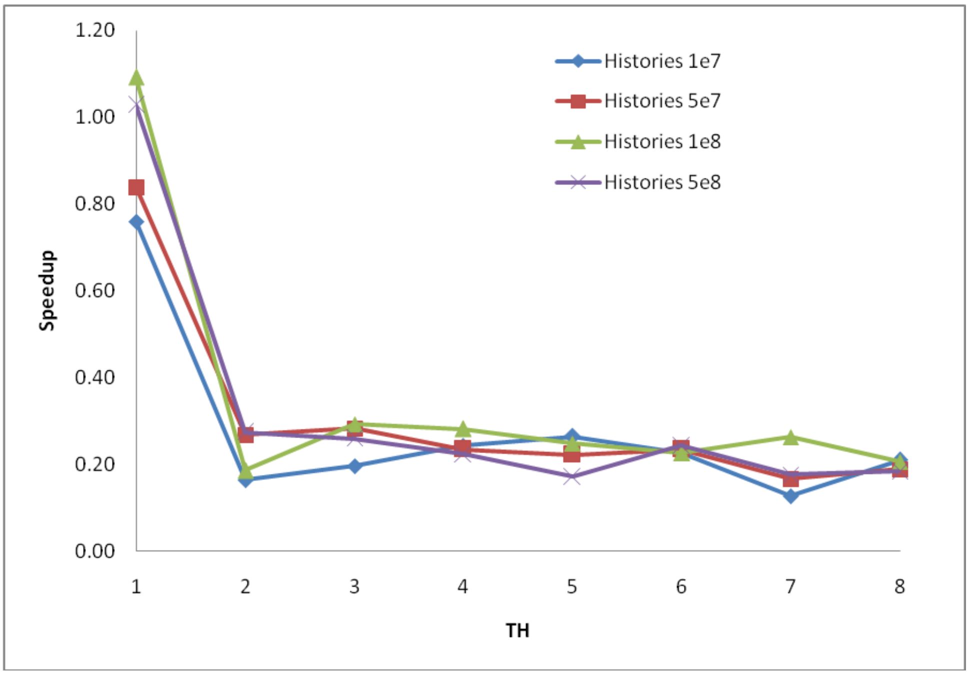

Fig. 2- Speedup Vs TH for different histories

3c) Figure 3 below shows the plot of efficiency of parallelization and number of threads. Efficiency of parallelization continues to drop with increasing number of threads, indicating increased overheads with number of threads.

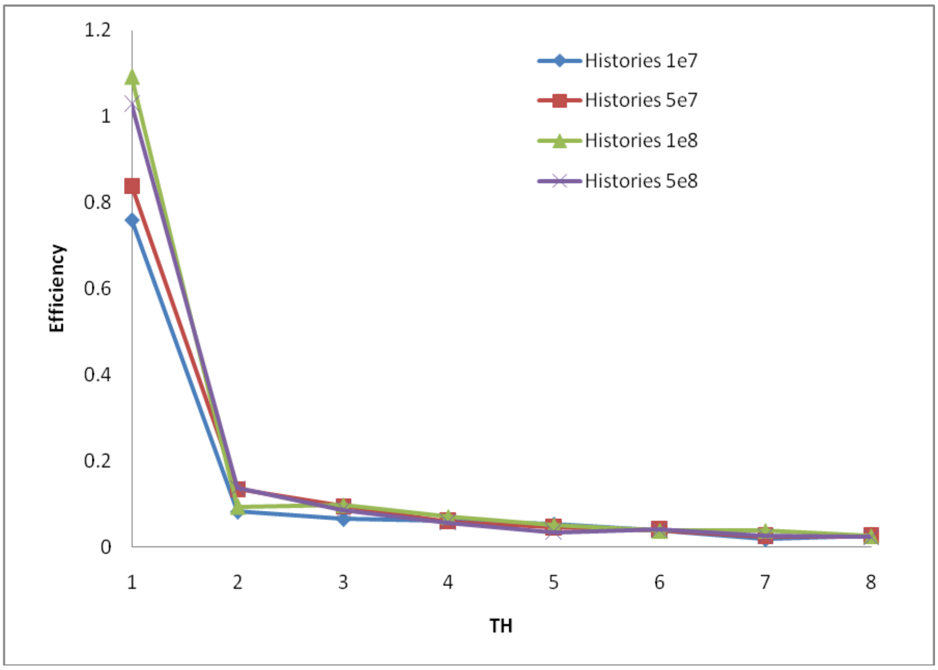

Fig3. Speedup Vs TH for different histories