HYBRID MPI/PTHREAD PARALLELIZATION OF THE WEIBULL DISTRIBUTION PARAMETERS ESTIMATOR

S.Duli [a], B.Krstajić [b]
[a] *Department of Computer Sciences, Faculty of Natural Sciences, University of Shkoder, Albania*
[b] *Department of Electrical Engineering, University of Montenegro, Podgorica, Montenegro*
Email : siditaduli@yahoo.com, bozok@ac.me

Abstract: *In this paper it is presented the hybrid programming implementation of MPI and Pthreads to estimate the parameters of Weibull distribution for a given data sample. Message passing communication is managed by MPI to achieved coarse grained parallelism between nodes, and shared memory communication is implemented by Pthreads. The aim of this article is to verify the speedup of the hybrid MPI/Pthread version, and to compare it to the techniques of parallelizing the code, the MPI and the POSIX, in estimating the Weibull parameters of large datasets.*

Keywords : parallel programming, Weibull distribution, MPI, Pthread

Introduction

Weibull Distribution is a statistical tool used to model the wind speed. The wind speed data collected in a weather station tend to have the form of a Weibull distribution. The shape of the graphics depends on the parameters of the distribution, which are estimated by the set of wind speed data. Once the Weibull distribution can be used to calculate the probability of a particular wind speed at a particular location, it can be used to work out the number of hours per year that certain wind speeds are likely to record and therefore the likely total power output of a wind turbine per year [1].

The principle of parallel programming is to divide the problem in smaller pieces and to distribute the calculations between many processors. In this way can be achieved a faster result than the same code executed in one processor. There are different ways of parallelization the code of this application. Two of them are the MPI and Pthread. The aim of this research is to represent a hybrid implementation of the parameter estimation of Weibull parameters.

Methods

Two methods used in implementing the parallel version of the estimation of Weibull parameters are MPI and Pthreads. The programs using MPI libraries may run on distributed-memory multicomputer, shared-memory multiprocessors, networks of workstations, and combinations of all of these. MPI is implemented on a great variety of machines, including those "machines" consisting of collections of other machines, parallel or not, connected by a communication network [2] . Meanwhile the Pthread library provides an interface to generate and interact with separate threads of execution within a program. This standard is defined by the IEEE and is available across nearly all variants of the UNIX operating systems [3] .The hybrid version MPI/Pthread is implemented by creating two threads for each process involved in the calculations.

The application takes a censored set of data, which might be ordered or unordered, taken from a sample of N data. It is given the location parameter of the Weibull distribution. The data sample and the location parameter are read from a file by each process that is created. The algorithm makes an estimation of the scale parameter and shape parameter of this distribution.

Also for the density function of Weibull distribution is calculated the mode, the mean and the variance for the given set of data. This procedure is repeated R times for different set of data, taken from this N data sample.

Results

Tests are performed with data samples of 10 000 000 values and 5 000 000 values. The Weibull parameters are calculated for 100 such samples, taken randomly from a bigger amount of data, from 15 000 000. For each of these samples, there are calculated the mode, the mean and the variance of the distribution. The performance is tested for different number of processors, beginning with the serial version with only one processor running a doing the calculations, up to twelve parallel processors doing the calculations. Tests are performed for each implementation, Pthread, MPI and Hybrid version with two threads.

No of Cores	Hybrid 2 threads	MPI version	Pthread Version
2	6421	6022	5960
4	3209	3098	2947
6	2195	2048	1998
8	1632	1567	1502
10	1315	1292	1213
12	1294	1220	1022

Table 1: Time of execution of different version for the sample containing 10 000 000 data.

Table 1 shows the time spent to estimate the Weibull parameters of a dataset containing 10 000 000 data. Tests are made for hybrid two Pthreads and one process, up to two Pthreads and 6 processes. For large dataset, both versions reduce the time spent to estimate the result.

Results show that the time spent to perform the calculations is reduced while increasing the number of cores. For this estimation, hybrid version executed with two threads and one process takes 6421 seconds. For the same dataset, this version takes 1294 second when executed with two threads and six processes. These results are compared with those of MPI version executed with the same number of processes as cores, and those of Pthread version executed with the same number of threads as cores. As it is shown in the table 1, for this range of dataset, the Pthread version needs less time to perform the estimation. Pthread version takes 5960 seconds to estimate the parameters while it is executed with two threads, and 1022 seconds while it is executed with twelve threads.

From table 1 we can conclude that MPI pure version nor hybrid MPI/Pthread is not better than Pthreads in terms of execution time[4].

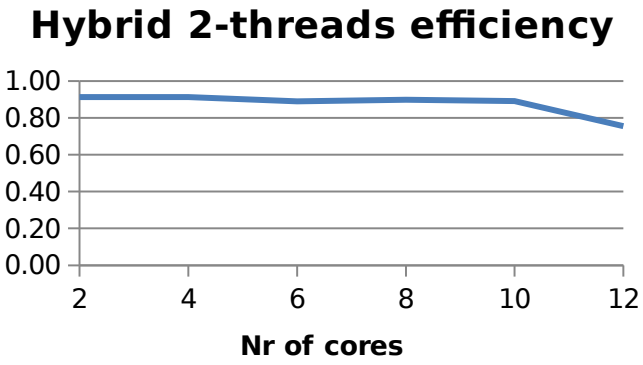

Fig 1. Efficiency hybrid MPI/Pthread version when data sample contains 10 000 000 data.

The figure 2 represents the efficiency of the hybrid MPI/Pthread version. Results show that the efficiency is relatively high when the number of cores is small. Adding new processes after the five process decreases the efficiency. Increasing the number of processes, the communication overhead becomes outstanding and the parallel performance degrades. The hybrid MPI/Pthreads programming mode could obtain a good parallel performance when the processes' number is larger by reducing the communication between processes.

The same implementation is executed applying different sample containing 5 000 000 data.

No of Cores	Hybrid 2 threads	MPI version	Pthread Version
2	3249	3078	2938
4	1618	1585	1471
6	1137	1120	997
8	819	806	741
10	765	752	625
12	729	712	560

Table 2: Time of execution of different parallel versions for a sample with 5 000 000 data.

Results show that hybrid version executed with two threads and one process takes 3249 seconds for estimating the Weibull parameters of a sample containing 5 000 000 data. For the same dataset, this version takes 729 second when executed with two threads and six processes. Results are compared with those of MPI version executed with the same number of processes as cores, and those of Pthread version executed with the same number of threads as cores. For this range of dataset, the Pthread version performs faster the calculations. Pthread version takes 2938 seconds to estimate the parameters while it is executed with two threads, and 560 seconds while it is executed with twelve threads.

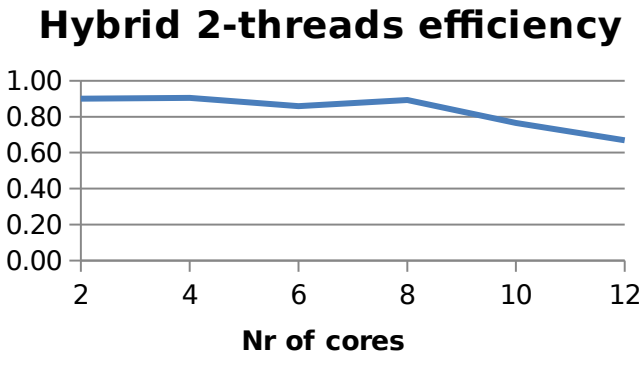

Fig 2. Efficiency hybrid MPI/Pthread version when data sample contains 5 000 000 data.

Figure 2 shows the graphical presentation of the efficiency of the hybrid 2 threads MPI/Pthreads version. Results show that for a small number of processes involved in the parallel calculations the efficiency is high. With increasing the number of processes the parallel performance begins to decrease. The graphic shows that the efficiency is still high when there are involved four processes. They are all combined with 2 threads each, distributed in eight cores, give a good performance.

Both figure 1 and figure 2 show that for a large dataset the efficiency stays high for bigger number of cores, respectively ten cores. For a smaller dataset, in figure 2, the efficiency is high only until the number of eight cores.

CONCLUSIONS

- The hybrid 2 threads MPI/Pthread might be as a technique of parallelization of estimation of Weibull distribution. The hybrid implementation performs the calculations in less time, as the number of processes involved is increased.
- The hybrid 2 threads MPI/Pthread needs more time to perform calculations than the pure MPI version. Also the hybrid is slower than the pure Pthread version. This is due to communication needed between processes.
- The efficiency of the hybrid 2 threads MPI/Pthread falls when the number of cores is higher than eight, for a sample 5 000 000. For the sample containing 10 000 000 data, the efficiency falls when the number of cores is higher than ten.
- As future work is analyzing the hybrid 4 threads MPI/Pthreads and comparing the results with this research.

REFERENCES

[1].http://www.reuk.co.uk/Wind-Speed-Distribution-Weibull.htm

[2].Y.AOYAMA, J. NAKANO : *Practical MPI Programming*, *International Technical Support Organization*,1999

[3].D. R BUTENHOF: *Programming with Pthread*, 1997

[4] Huailiang Xuan, Yue Hu, Weiqin Tong, Zhixun Gong and Yan Hou, "*Hybrid programming implementation of BiCG method on SMP cluster architecture*", IEEE/ACIS 11 International Conference on Computer and Information Science, 2012