CPAR project 1
Performance Evaluation
20/02/2018

In this project we will study the effect on the processor performance of the memory hierarchy when accessing large amounts of data. The product of two matrices will be used for this study.

1. Implement the basic matrix multiplication algorithm in C/C++, i.e. the algorithm that multiplies one line of the first matrix by each column of the second matrix. Implement the same algorithm in other programming languages, such as JAVA, C#, Python, etc.

Register the processing time for the several implementations of the algorithm, for input matrices from 500x500 to 3000x3000 elements with increments in both dimensions of 500.

Use the Performance API (PAPI) and Tiptop to collect relevant performance indicators of the program execution.

2. Implement a C/C++ version that multiplies an element from the first matrix by the correspondent line of the second matrix.

Register the processing time for the two versions of the algorithm, for input matrices from 500x500 to 3000x3000 elements with increments in both dimensions of 500.

Register the processing time from 5000x5000 to 10000x10000 with intervals of 1000.

Use PAPI and Tiptop to collect relevant performance indicators of the program execution.

3. Implement a parallel version of the two versions of the algorithm in C/C++ using OpenMP.

Register the processing time for input matrices from 500x500 to 3000x3000 elements with increments in both dimensions of 500 and using from 1 to 8 threads.

Use PAPI and Tiptop to collect relevant performance indicators of the program execution.

OUTCOMES

Write a report of up to 6 pages explaining the algorithm versions and analyzing the results obtained. Justify the performance parameters selected and use them to evaluate and compare the versions implemented.

To be delivered on: 13/03/2018

Parameters for Report Evaluation:

- Problem description and algorithms explanation;
- Performance metrics and evaluation methodology;
- Results and analysis;
- Conclusions;
- Writing quality.