PMC Lecture 07

Gianmaria Romano

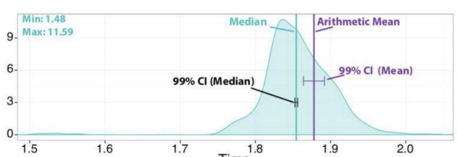
22 October 2025

Chapter 2

Message passing interfaces

2.1 Evaluating the performance of parallel applications

Evaluating the performance of a parallel system can be tricky because, even though it is possible to estimate the running time in terms of the slowest process, this metric tends to be unreliable as processes might start at different times.
For this reason, it is a good practice to implement a barrier that makes sure that processes start at approximately the same time.
Keep in mind, however, that running an application just once cannot provide a good evaluation due to eventual interference from other applications or from the communication network, so it is a good practice to analyse the distribution of the running times.

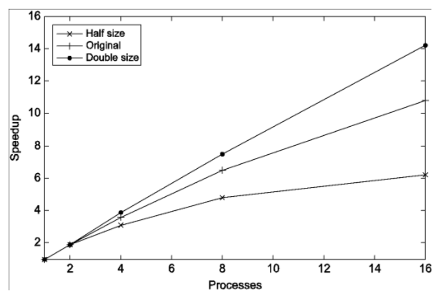


2.1.1 The speed-up factor of parallelism

Assuming $T_{seq}(n)$ denotes the running time of a sequential application on a problem of size n while $T_{par}(n, p)$ denotes the running time of a parallel application that uses p processes to solve the same problem, it is possible to define the speed-up factor of the parallel application as the performance improvement that is obtained by parallelizing the application, which can be measured as follows:

$$S(n,p) = \frac{T_{seq}(n)}{T_{par}(n,p)}$$

Ideally, parallelizing an application should provide a linear speed-up, which is achieved whenever $S(n,p) = p$, although, generally speaking, an application tends to benefit from parallelism only for larger inputs.



In addition, it is possible to evaluate the performance of a parallel application in terms of its efficiency, which measures how well a program exploits its resources through the following measurement:

$$E(n,p) = \frac{S(n,p)}{p} = \frac{T_{seq}(n)}{pT_{par}(n,p)}$$

Ideally, one should achieve $E(n,p) = 1$, although, generally speaking, this condition is unfeasible to achieve due to the implementation trade-offs for parallel systems.

2.1.2 The scalability of a parallel application

From the definition of speed-up, it is also possible to define the scalability of a parallel application as the performance improvement that is obtained by going from running the application using one process to running it using p processes, which can be measured as follows:

$$S(n,p) = \frac{T_{par}(n,1)}{T_{par}(n,p)}$$

Overall, an application is said to be "strong scaling" if fixing the problem size and increasing the number of processes keeps high efficiency.
On the other hand, an application is said to be "weak scaling" if it increases the number of processes only to deal with larger problems.