

# Computação paralela / Computação Avançada (2016/2017)

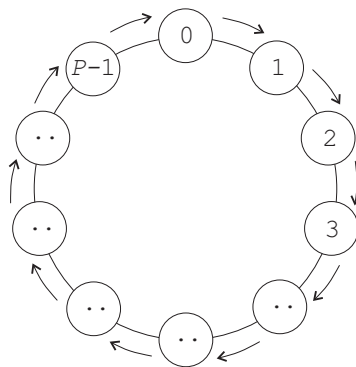
## 2º trabalho para casa

Prazo de entrega: 5/12/2016

Considere a multiplicação de uma matriz quadrada  $A$  por um vector  $\mathbf{x}$ ,

$$\mathbf{y} = A\mathbf{x} .$$

1. Faça um programa série que calcule  $\mathbf{y}$  para uma dimensão variável  $n$ , sendo todas as variáveis de dupla precisão. Estabeleça uma fórmula geral para inicializar os elementos da matriz  $A$  e do vector  $\mathbf{x}$ , que servirá também para o programa paralelo descrito a seguir. Use uma rotina de sistema para calcular o tempo que demora a calcular  $\mathbf{y}$  para vários valores de  $n$  (considere valores de  $n$  até 10000).
2. Elabore agora programas em MPI para fazer o mesmo cálculo, distribuindo a alocação da matriz  $A$  e do vector  $\mathbf{x}$  pelos  $p$  processadores, ou seja, alocando  $n/p$  linhas de  $A$  e de  $\mathbf{x}$  a cada processo, o que implica que considere valores de  $n$  que sejam divisíveis por  $p$ . Considere duas topologias de comunicação (ver secção 13.1 da ref. [1]):
  - tipo “anel” (ver figura em baixo), em que cada processo de “rank” `myrank` recebe a parte do vector  $\mathbf{x}$  que processo anterior (`myrank-1`) tem e envia a sua parte ao processo seguinte (`myrank+1`). O processo é repetido até que cada um dos processos fique com toda a informação. Procure sobrepor ao máximo os processos de cálculo e de comunicação, minimizando também a alocação de memória em cada processo.
  - tipo “allgather” (use rotina MPI `MPI_ALLGATHER/MPI_ALLgather`) para comunicar de todos para todos as partes do vector  $\mathbf{x}$ .
3. Usando a rotina `MPI_WTIME()` (FORTRAN) ou `MPI_Wtime()` (C), faça gráficos do “speed-up” (tempo de CPU série/tempo CPU paralelo) para três valores de  $n$  (1000, 5000, 10000), para as duas topologias e para  $p = 2, 4, 8$ . Comente os resultados.



## Referências

- [1] Peter Pacheco, Parallel Programming with MPI, Morgan Kaufmann Publishers.

# Computação Paralela (2016/2017)

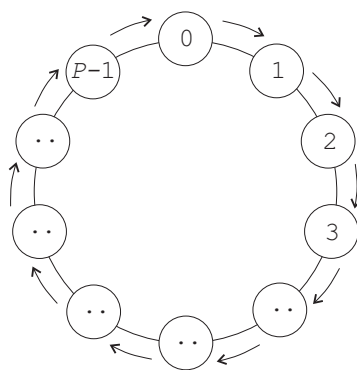
## 2nd homework

**Deadline: 5/12/2016**

Consider the multiplication of a square matrix  $A$  by a vector  $\mathbf{x}$ ,

$$\mathbf{y} = A\mathbf{x} .$$

1. Write a serial program to calculate  $\mathbf{y}$  for a variable dimension  $n$ , all variables being double precision. Devise a general formula to set the elements of matrix  $A$  and of vector  $\mathbf{x}$ , which shall be used also for the parallel program described below. Use a system call to obtain the time it takes to calculate  $\mathbf{y}$  for several values of  $n$  (consider values of  $n$  up to 10000).
2. Write MPI programs to perform the same calculation by equally allocating the lines of matrix  $A$  and vector  $\mathbf{x}$  to the  $p$  processors, which means that  $n$  must be a multiple of  $p$ . Consider two communications topologies (see section 13.1 of ref. [1]):
  - “ring” type (see figure below), in which each process with rank `myrank` receives the part of vector  $\mathbf{x}$  that the previous process (`myrank-1`) currently has and pass its own part to the next process (`myrank+1`). The process is repeated until each process gets the whole information. Try to overlap as much as possible the calculations and the communications, also minimizing the memory allocation in each process;
  - “allgather” type (using the MPI routine `MPI_ALLGATHER/MPI_Allgather`) to communicate from all to all processors the parts of vector  $\mathbf{x}$ .
3. Using the call `MPI_WTIME()` (FORTRAN) or `MPI_Wtime()` (C), do plots of the speed-up (CPU serial time/CPU parallel time) for 3 values of  $n$  (1000, 5000, 10000), for the two topologies and for  $p = 2, 4, 8$ . Comment the results.



## Referências

- [1] Peter Pacheco, Parallel Programming with MPI, Morgan Kaufmann Publishers.