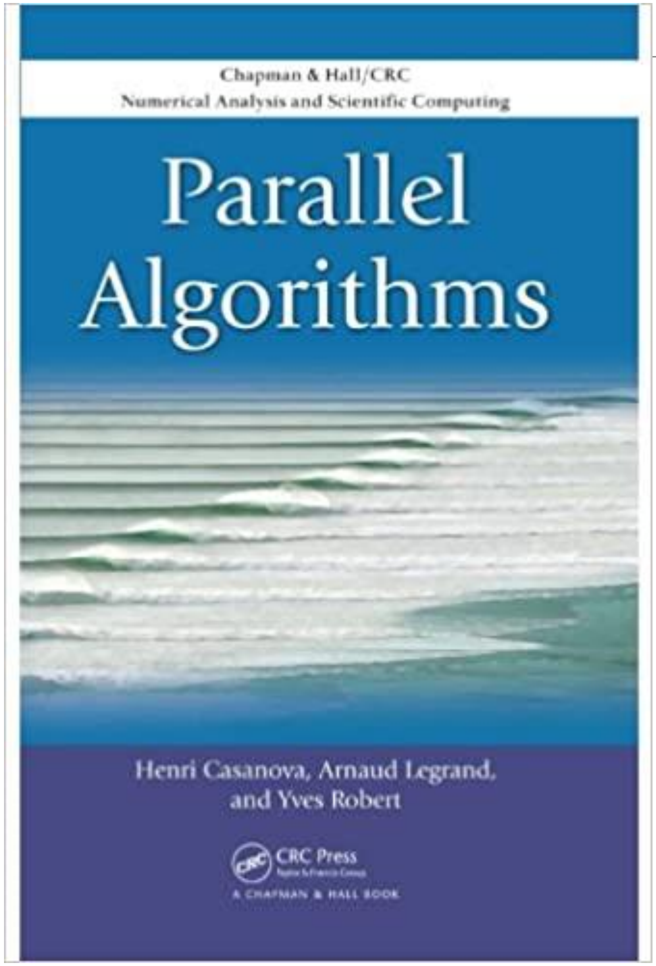
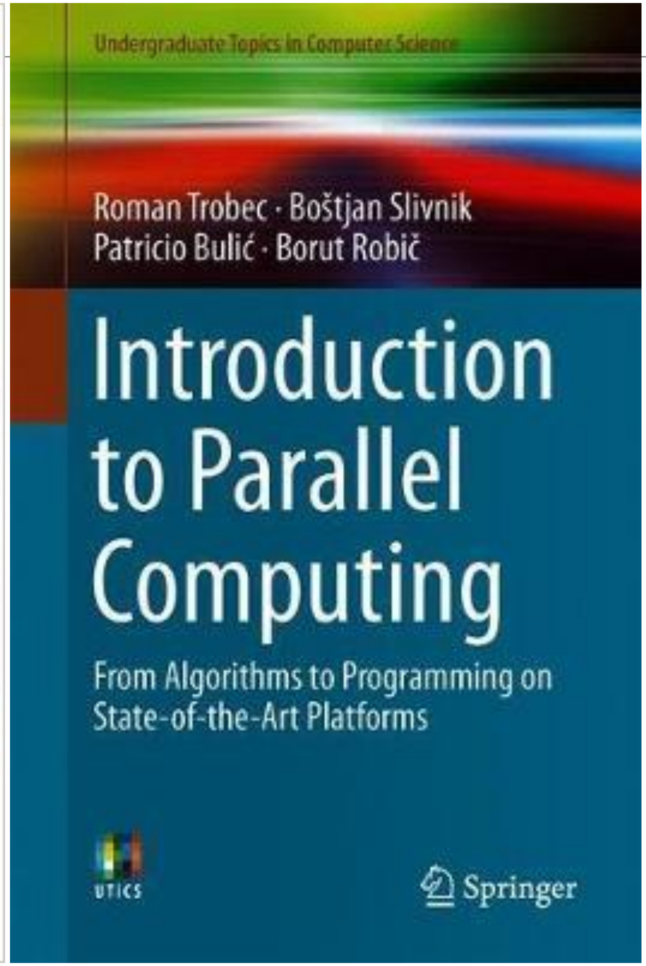
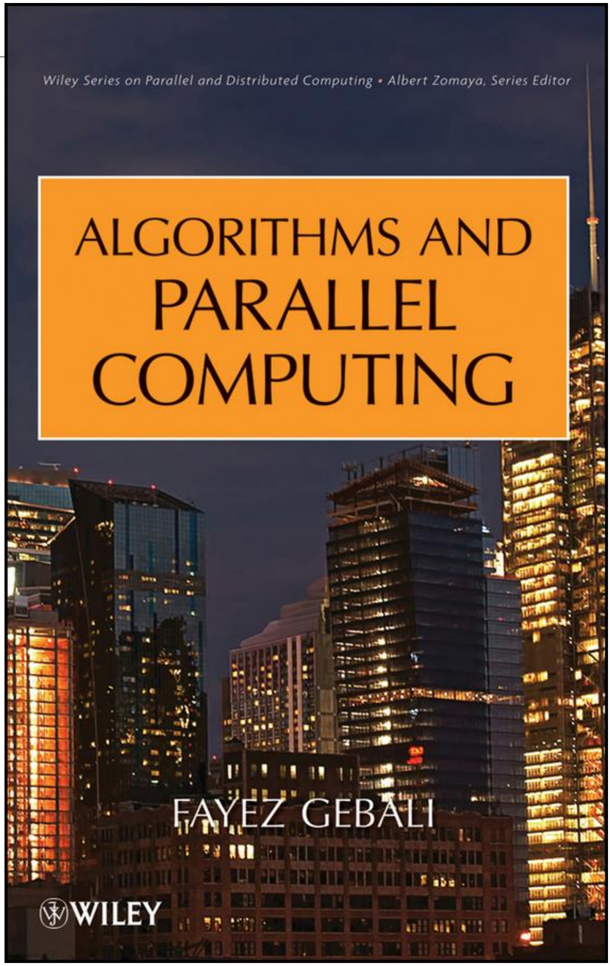
Parallel Programming

Lec 6

Books

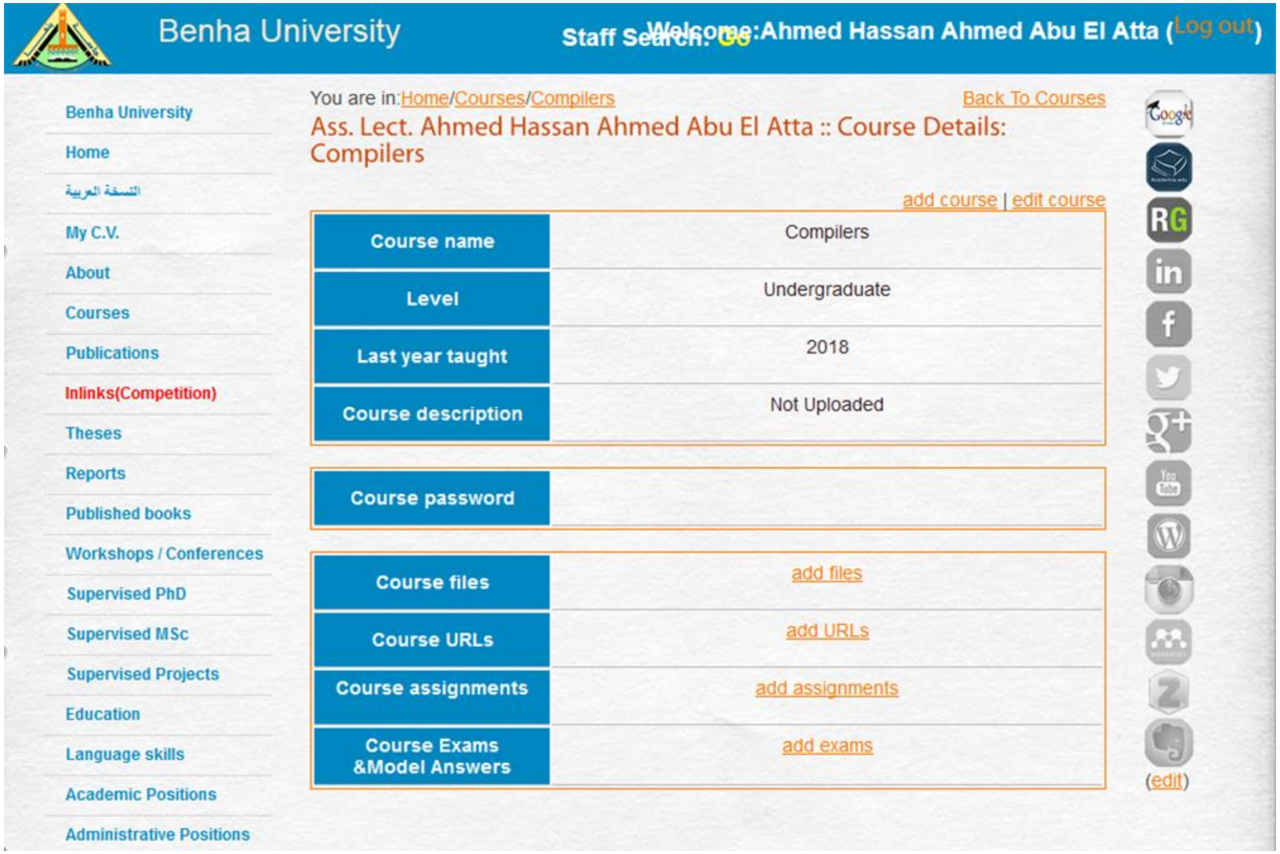

Chapman & Hall/CRC
Numerical Analysis and Scientific Computing
Parallel Algorithms
Henri Casanova, Arnaud Legrand, and Yves Robert
CRC Press
A CHAPMAN & HALL BOOK


Undergraduate Topics in Computer Science
Roman Trobec · Boštjan Slivnik
Patricio Bulić · Borut Robič
Introduction to Parallel Computing
From Algorithms to Programming on State-of-the-Art Platforms
UTICS
Springer


Wiley Series on Parallel and Distributed Computing • Albert Zomaya, Series Editor
ALGORITHMS AND PARALLEL COMPUTING
FAYEZ GEBALI
WILEY

PowerPoint

http://www.bu.edu.eg/staff/ahmedaboalatah14-courses/14779



Matrix-Vector Multiplication

Problem Statement

The result of multiplying the matrix A of order $m \times n$ by vector b, which consists of n elements, is the vector c of size m

Each i^{th} element of which is the result of inner multiplication of i^{th} matrix A row (let us denote this row by a_i) by vector b:

$$c_i = (a_i, b) = \sum_{j=0}^{n-1} a_{ij} b_j \ , 0 \le i \le m-1$$

Multiplying a square matrix by a vector (**Sequential algorithm**)

Simply a series of dot products

Input: Matrix A[m][n]

Vector B[n]

Output: C[m]

```
for ( i = 0; i < m; i++ )
    C[i] = 0;
    for ( j = 0; j < n; j++ )
        C[i] += A[i][j] * B[j];
```

Multiplying a square matrix by a vector (Sequential algorithm)

Inner loop requires n multiplications and n - 1 additions

Complexity of inner loop is O(n)

There are a total of m dot products

Overall complexity: O(mn)

$O(n^2)$ for a square matrix

Parallel Methods for Matrix-Vector Multiplication

Data Distribution

The repetition of the same computational operations for different matrix elements is typical of different matrix calculation methods.

In this case we can say that there exist data parallelism.

As a result, the problem to parallelize matrix operations can be reduced in most cases to matrix distributing among the processors of the computer system.

The most general and the most widely used matrix distribution methods consist in partitioning data into stripes (vertically and horizontally) or rectangular fragments (blocks).

Data Distribution

1. Rowwise block striping

- Divide matrix elements into group of rows
- Each process responsible for a contiguous group of m/p rows

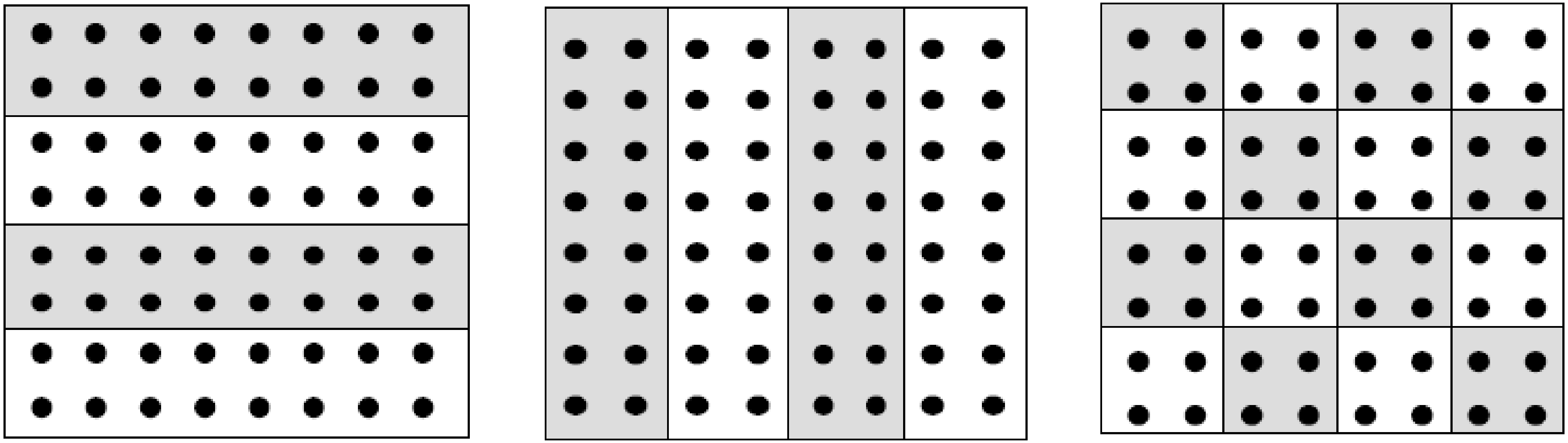
2. Columnwise block striping

- Divide matrix elements into group of columns
- Each process responsible for a contiguous group of n/p columns

3. Checkerboard block decomposition

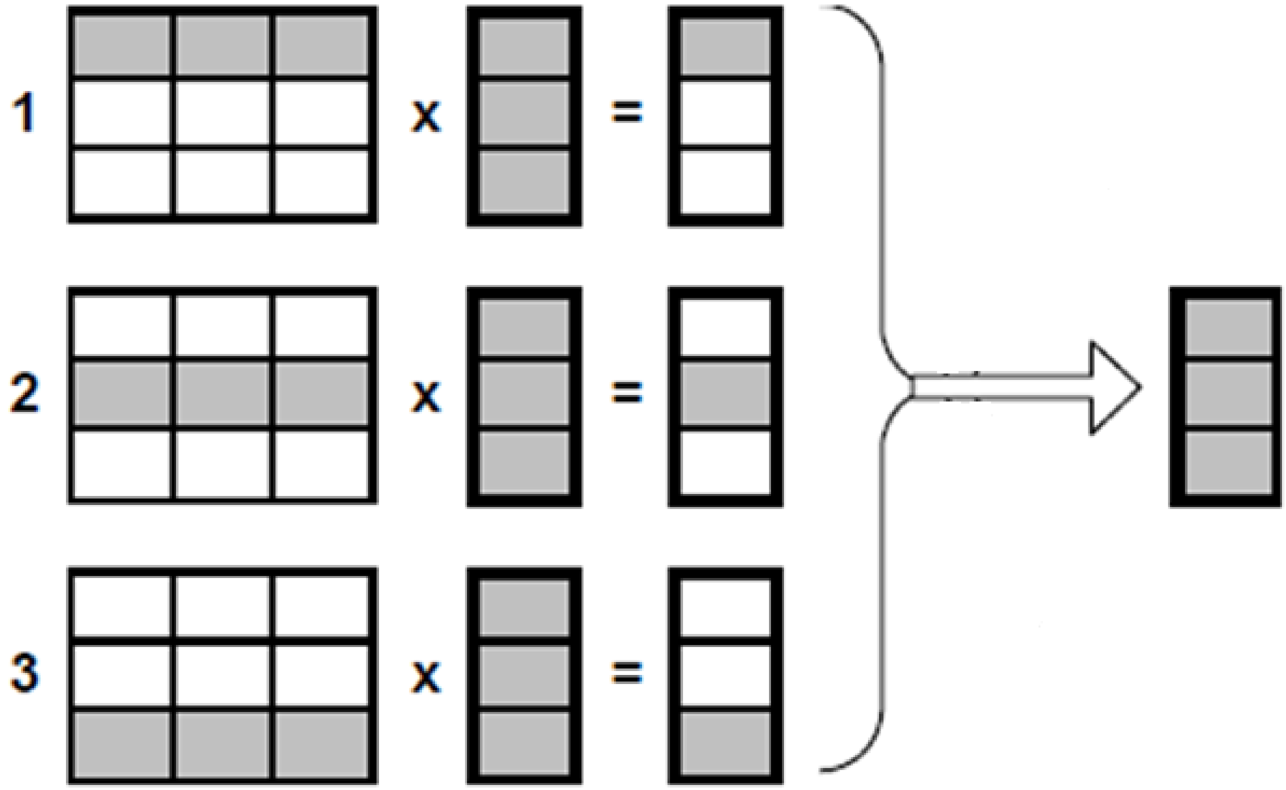
- Form a virtual grid
- Matrix is divided into 2-D blocks aligning with the grid
- Let the grid have r rows and c columns
- Each process responsible for a block of matrix containing at most m/r rows and n/c columns

Data Distribution



Matrix-Vector Multiplication in Case of Rowwise Data Decomposition

Matrix-Vector Multiplication in Case of Rowwise Data Decomposition



Matrix-Vector Multiplication in Case of Rowwise Data Decomposition

If this case, the operation of inner multiplication of a row of the matrix A and the vector b can be chosen as the basic computational subtask.

To execute the basic subtask of inner multiplication the processor must contain the corresponding row of matrix A and the copy of vector b.

After computation completion each basic subtask determines one of the elements of the result vector c.

Matrix-Vector Multiplication in Case of Rowwise Data Decomposition

Rowwise Parallel Matrix-Vector(A,B,C)

```
For i = 0 to n-1 do in parallel
    C[i] = 0
    For j = 0 to n-1 do
        C[i] = C[i] + A[i, j] * B[j]
    End for
End parallel
```

Matrix-Vector Multiplication in Case of Rowwise Data Decomposition

If matrix A is square (m=n), the sequential algorithm of matix-vector multiplication has the complexity $T_s=O(n^2)$.

In case of parallel computations each processor performs multiplication of only a part (stripe) of the matrix A by the vector b.

The size of these stripes is equal to n/p rows (or one row if p = n).

In case of computing the inner product of one matrix row by a vector, it is necessary to perform the n multiplications and (n-1) additions.

$T_p(n) = O(n^2/p)$, when $p = n$; $T_p(n) = O(n)$

$S_p(n) = \frac{n^2}{n^2/p} = p$, when $p = n$; $S_p(n) = O(n)$ $\quad E_p(n) = \frac{n^2}{p * n^2/p} = 1$

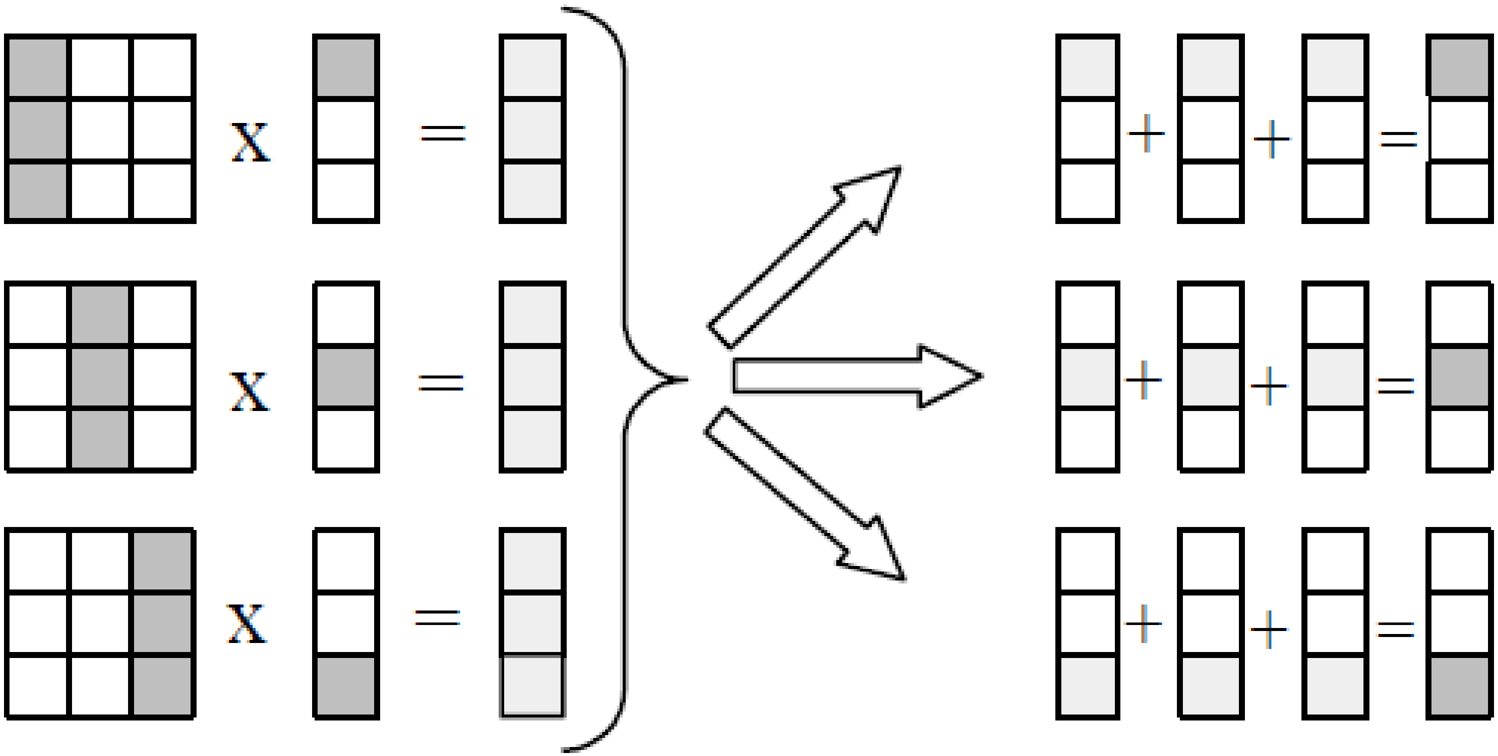
Matrix-Vector Multiplication in Case of Columnwise Data Decomposition

Matrix-Vector Multiplication in Case of Columnwise Data Decomposition

In case of columnwise matrix decomposition the operation of multiplying a column of matrix A by one of the vector b elements may be chosen as the basis computational subtask.

As a result to perform computations each basic subtask i, $0 \le i < n$, must contain the i^{th} column of matrix A and the i^{th} elements b_i and c_i of vectors b and c.

Matrix-Vector Multiplication in Case of Columnwise Data Decomposition



Matrix-Vector Multiplication in Case of Columnwise Data Decomposition

At the starting point of the parallel algorithm of matrix-vector multiplication each basic task i carries out the multiplication of its matrix A column by element b_i. As a result, vector c'_i (the vector of intermediate results) is obtained in each subtask.

each basic subtask i, $0 \le i < n$, will contain n partial values $c'_i(j)$, $0 \le j < n$. Element c_i of the result vector c is determined after the addition of the partial values

Matrix-Vector Multiplication in Case of Columnwise Data Decomposition

Columnwise Parallel Matrix-Vector(A,B,C)

```
For j = 0 to n-1 do in parallel
    For i = 0 to n-1 do
        C'[i][j] = A[i][j] * B[j]
    End for
End parallel
For i = 0 to n-1 do in parallel
    C[i] = 0
    For j = 0 to n-1 do
        C[i] = C[i] + C'[i][j]
    End for
End parallel
```

Matrix-Vector Multiplication in Case of Columnwise Data Decomposition

If matrix A is square (m=n), the sequential algorithm of matix-vector multiplication has the complexity $T_s=O(n^2)$.

In case of parallel computations each processor performs multiplication of only a part (stripe) of the matrix A by the vector b.

The size of these stripes is equal to n/p rows (or one row if p = n).

In case of computing the inner product of one matrix row by a vector, it is necessary to perform the n multiplications and (n-1) additions.

$T_p(n) = O(n^2/p)$, when $p = n$; $T_p(n) = O(n)$

$S_p(n) = \frac{n^2}{n^2/p} = p$, when $p = n$; $S_p(n) = O(n)$ $\quad E_p(n) = \frac{n^2}{p * n^2/p} = 1$