HUMAN CAPITAL
NATIONAL COHESION STRATEGY

EUROPEAN UNION
EUROPEAN
SOCIAL FUND

High Performance Computing

Lecture 4 - Parallel Linear Algebra

Felicja Okulicka-Dłużewska

F.Okulicka@mini.pw.edu.pl

Warsaw University of Technology

Faculty of Mathematics and Information Science

WARSAW UNIVERSITY OF TECHNOLOGY
DEVELOPMENT PROGRAMME

Go to full-screen mode now by hitting CTRL-L

Overview

- **Introduction**
- Gaussian elimination
  - Parallelization by vector operations
  - Blas implementation
- Parallel backward substitution

Introduction

Many problems need to solve linear equations. Examples of matrices from the engeenering problems can be found on Matrix Market:

www.MatrixMarket.com

Introduction

Several methods for solving the set of linear equations have been introduced during years. Generally they are divided on two classes:

- Direct methods: for example Gaussian elimination, LU decomposition,
- Iterative methods:
  - Stationary iterative methods: Jacobi, Gauss-Seidel, SOR, SSOR ,
  - Krylov subspace methods: CG (Conjugate Gradient), GMRES (Generalized Minimum Residuum), Steepest Descent.

Introduction

The most popular direct method is LU decomposition. It is equivalent to Gaussian elimination. Both lead to the systems with triangular matrices, which can be solved by backward or forward elimination.

Iterative methods create the sequence of the approximate solutions which should converge to the exact solution of the problem.

Introduction

Two types of the parallelization of linear algebra algorithms are used:

- Parallelization of the matrix operations:
  - vector calculations,
  - block calculations .
- Divide and Conquer Algorithm (for example Block Cyclic Reduction)

Introduction

Linear algebra algorithms make computation on matrices and can be nicely parallelized in "natural" and "intuitive" way.
To solve the system of linear equations in parallel by direct methods two aproaches are in use:

- parallelization by the application of the parallel operations (vector and matrix operations)
- division of the data on the separate blocks in the separate memories to apply the parallel block algorithms

Introduction

In direct methods the parallel vectors calculations are applied due to for instance changing the order of the operations. The vector operations can be implemented on the vector-computers or on the shared memory machines using BLAS.

The block algorithms allow to spit the data on distributed memories and to make part of the computation concurrently by each processor on the possessed data. The synchronization is important in the implementation.

Introduction

Divide and Conquer Algorithm has three main steps:

- Divide the problem into two or mode independent subproblems of the same type and structure
- Conquer - i.e. solve the divided problems
- Reconstruct the solution of the original problem from the solutions of the subproblems

Introduction

Divide and Conquer is a form of recursion.

Block cyclic reduction is the example of both: block and Divide and Conquer methods.

Overview

- Introduction
- **Gaussian elimination**
  - Parallelization by vector operations
  - Blas implementation
- Parallel backward substitution

Gaussian Elimination

Consider the system of linear equations

$$Ax = b$$

$Ax = b$

$$\begin{pmatrix} a_{1,1} & \dots & a_{1,k} & \dots & a_{1,n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{k,1} & \dots & a_{k,k} & \dots & a_{k,n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n,1} & \dots & a_{n,k} & \dots & a_{n,n} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ \dots \\ x_k \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ \dots \\ b_k \\ \dots \\ b_n \end{pmatrix}$$

Gaussian Elimination

At the 1-st stage x_1 is eliminated from equations $i = 2, 3, ..., n$ by the subtraction of $a_{ik}^{(k-1)}/a_{11}$ times equation 1 from each of these equations:

$$A^{(1)}x = b^{(1)}$$

where

$A^{(1)}$

$$\begin{pmatrix} a_{11} & a_{1,2} & \dots & a_{1,k} & a_{1,k+1} & \dots & a_{1,n} \\ 0 & a^{(1)}_{2,2} & \dots & a^{(1)}_{2,k} & a^{(1)}_{2,k+1} & \dots & a^{(1)}_{k-1,n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & a^{(1)}_{k,2} & \dots & a^{(1)}_{k,k} & a^{(1)}_{k,k+1} & \dots & a^{(1)}_{k,n} \\ 0 & a^{(1)}_{k+1,2} & \dots & a^{(1)}_{k+1,k} & a^{(1)}_{k+1,k+1} & \dots & a^{(k-1)}_{k+1,n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & a^{(1)}_{n-1,2} & \dots & a^{(1)}_{n-1,k} & a^{(1)}_{n-1,k+1} & \dots & a^{(1)}_{n-1,n} \\ 0 & a^{(1)}_{n,2} & \dots & a^{(1)}_{n,k} & a^{(1)}_{n,k+1} & \dots & a^{(1)}_{n,n} \end{pmatrix}$$

Gaussian Elimination

At the k-th stage x_k is eliminated from equations $i = k+1, k+2, ..., n$ by the subtraction of $a_{ik}^{(k-1)}/a_{kk}^{(k-1)}$ times equation k from each of these equations:

$$A^{(k-1)}x = b^{(k-1)}$$

where

$A^{(k-1)}$

$$\begin{pmatrix}
a_{11} & \dots & a_{1,k-1} & a_{1,k} & a_{1,k+1} & \dots & a_{1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & a^{(k-2)}_{k-1,k-1} & a^{(k-2)}_{k-1,k} & a^{(k-2)}_{k-1,k+1} & \dots & a^{(k-2)}_{k-1,n} \\
0 & \dots & 0 & a^{(k-1)}_{k,k} & a^{(k-1)}_{k,k+1} & \dots & a^{(k-1)}_{k,n} \\
0 & \dots & 0 & a^{(k-1)}_{k+1,k} & a^{(k-1)}_{k+1,k+1} & \dots & a^{(k-1)}_{k+1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & 0 & a^{(k-1)}_{n-1,k} & a^{(k-1)}_{n-1,k+1} & \dots & a^{(k-1)}_{n-1,n} \\
0 & \dots & 0 & a^{(k-1)}_{n,k} & a^{(k-1)}_{n,k+1} & \dots & a^{(k-1)}_{n,n}
\end{pmatrix}$$

Gaussian Elimination

$$b^{(k-1)} = \begin{pmatrix} b_1 & b_2^{(1)} & \dots & b_k^{(k-1)} & \dots & b_{n-1}^{(k-1)} & b_n^{(k-1)} \end{pmatrix}^T$$

where

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{ik}^{(k-1)}$$

$$b_i^{(k)} = b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)}$$

Gaussian Elimination

After $n-1$ such elimination stages :

$$A^{(n-1)}x = b^{(n-1)}$$

Where $A^{(n-1)}$ is an upper triangular matrix.

$A^{(n-1)}$

$$\begin{pmatrix}
a_{11} & \dots & a_{1,k-1} & a_{1,k} & \dots & a_{1,n-1} & a_{1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & a^{(k-2)}_{k-1,k-1} & a^{(k-2)}_{k-1,k} & \dots & a^{(k-2)}_{k-1,n-1} & a^{(k-2)}_{k-1,n} \\
0 & \dots & 0 & a^{(k-1)}_{k,k} & \dots & a^{(k-1)}_{k,n-1} & a^{(k-1)}_{k,n} \\
0 & \dots & 0 & 0 & \dots & a^{(k)}_{k+1,n-1} & a^{(k)}_{k+1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & 0 & \dots & 0 & a^{(n-2)}_{n-1,n-1} & a^{(n-2)}_{n-1,n} \\
0 & \dots & 0 & 0 & \dots & 0 & a^{(n-1)}_{n,n}
\end{pmatrix}$$

Computational cost

At the $k-th$ stage of the elimination process:

- compute $(n-k)$ multipliers (1 flop each)
- modify $(n-k)$ right-hand side components (each component involves 2 flops)
- modify the $(n-k)\times(n-k)$ submatrix, which requires $2(n-k)^2$ flops.

Computational cost

The overall cost of the $n-1$ elimination steps is:

$$\sum_{k=1}^{n-1}(3(n-k)+2(n-k)^2)$$

$$=\sum_{k=1}^{n-1}(3k+2k^2)=\frac{2n^2}{3}+O(n^2)$$

Thus:

Forward elimination is an $O(n^3)$ process on $O(n^2)$ data.

Gaussian Elimination

```
 Void Gauss ( int n; float
a[][n],b[n],x[] )
{ int i,j,k ;
for (k=0;k=n-2;k++)
   {
     for (i=k+1;i=n-1;i++)
       a[i,k]=a[i,k]/a[k,k];
     for (j=k+1;j=n-1;j++)
       a[i,j]=a[i,j]-a[i,k]*a[k,j];
     b[i]= b[i]-a[i,k]*b[k];
  } ;
}
```

Gaussian Elimination

In the $k-th$ stage the calculation proceeds on the matrix:

$$\begin{pmatrix}
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & 0 & a_{k,k}^{(k-1)} & a_{k,k+1}^{(k-1)} & \dots & a_{k,n}^{(k-1)} & a_{k,n}^{(k-1)} \\
\dots & 0 & a_{k+1,k}^{(k-1)} & a_{k+1,k+1}^{(k-1)} & \dots & a_{k+1,n}^{(k-1)} & a_{k+1,n}^{(k-1)} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
\dots & 0 & a_{n-1,k}^{(k-1)} & a_{n-1,k+1}^{(k-1)} & \dots & a_{n-1,n-1}^{(k-1)} & a_{n-1,n}^{(k-1)} \\
\dots & 0 & a_{n,k}^{(k-1)} & a_{n,k+1}^{(k-1)} & \dots & a_{n,n-1}^{(k-1)} & a_{n,n}^{(k-1)}
\end{pmatrix}$$

Gaussian Elimination

In the **parallel algorithm** in the first step the whole vector of multipliers is calculated in one step:

$$mn^{(k)} = \left(0 \quad \ldots \quad 0 \quad \frac{a_{k+1,k}^{(k-1)}}{a_{kk}^{(k-1)}} \quad \frac{a_{k+2,k}^{(k-1)}}{a_{kk}^{(k-1)}} \quad \ldots \quad \frac{a_{n-1,1}^{(k-1)}}{a_{kk}^{(k-1)}} \quad \frac{a_{n,1}^{(k-1)}}{a_{kk}^{(k-1)}} \right.$$

```
 // compute the multipliers
// for equation k and overwrite
// the k-th column of a - Blas
for (l=k+1;l=n;l++)
    a[l,k]=a[l,k]/a[k,k];
```

Gaussian Elimination

Then the matrix -vector calculation is done. We multiply two vectors and the result (the matrix) is subtracted from the submatrix $A(k+1:n, k+1:n)$

$$A^{(k)} = A^{(k-1)} - mn^{(k)} * A^{(k-1)}_{k,*}$$

```
// modify the submatrix

for (i=k+1;i=n-1;i++)
    for (j=k+1;j=n-1;j++)
        a[i,j]=a[i,j]-a[i,k]*a[k,j];
```

Gaussian Elimination

Vector b can be calculated in parallel in the loop:

```
 for (i=k+1;i=n-1;i++)
// modify the i-th component
// of the rigth-hand side
b[i]= b[i]-a[i,k]*b[k];
```

Algorithm 1

```
Void Gauss1 (int n;float
a[][n],b[n],x[])
{ int i,j,k ;
for (k=0;k=n-2;k++)
   { for (i=k+1;i=n-1;i++)
     a[i,k]=a[i,k]/a[k,k];
   for (i=k+1;i=n-1;i++)
     for (j=k+1;j=n-1;j++)
       a[i,j]=a[i,j]-a[i,k]*a[k,j];
   for (i=k+1;i=n-1;i++)
     b[i]= b[i]-a[i,k]*b[k];
   };
}
```

Blas implementation

Above loops can be implemented using BLAS library.

Example

$$\begin{pmatrix} 4 & 1 & 0 & 1 \\ -4 & 2 & 2 & -1 \\ 0 & 0 & 3 & 1 \\ -4 & -1 & -3 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7 \\ 3 \\ 7 \\ -9 \end{pmatrix}$$

Example

Step 1

We calculate the vector of multipliers by dividing the first column by the element on the diagonal $a_{1,1}$:

$$mn^{(1)} = \frac{1}{4}\begin{pmatrix} 0 \\ -4 \\ 0 \\ -4 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \\ 0 \\ -1 \end{pmatrix}$$

Example

We eliminate the first variable from the equations $2, 3, 4$:

$$A^{(1)} = A^{(0)} - mn^{(1)} * A^{(0)}_{1,*} =$$

$$\begin{pmatrix} 4 & 1 & 0 & 1 \\ -4 & 2 & 2 & -1 \\ 0 & 0 & 3 & 1 \\ -4 & -1 & -3 & 3 \end{pmatrix} - \begin{pmatrix} 0 \\ -1 \\ 0 \\ -1 \end{pmatrix} \begin{pmatrix} 4 & 1 & 0 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 1 & 0 & 1 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & -3 & 4 \end{pmatrix}$$

Example

We calculate the new vector of right side of the equation:

$$b^{(1)} = \begin{pmatrix} 7 \\ 3 \\ 7 \\ -9 \end{pmatrix} - b_1^{(0)} \cdot mn^{(1)} = \begin{pmatrix} 7 \\ 3 \\ 7 \\ -9 \end{pmatrix} - b_1^{(0)} \begin{pmatrix} 0 \\ -1 \\ 0 \\ -1 \end{pmatrix}$$

$$= \begin{pmatrix} 7 \\ 3 \\ 7 \\ -9 \end{pmatrix} - 7 \begin{pmatrix} 0 \\ -1 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ -2 \end{pmatrix}$$

Example

Step 2

We calculate the vector of multipliers by dividing the second column by the element on the diagonal $a_{2,2}$:

$$mn^{(2)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

The vector of multipliers is equal zero- the matrix $A^{(2)} = A^{(1)}$ and the vector $b^{(2)} = b^{(1)}$.

Example

Step 3

We calculate the vector of multipliers by dividing the third column by the element on the diagonal $a_{3,3}$:

$$mn^{(3)} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

Example

We eliminate the variable x_3 from the equation 4:

$$\begin{pmatrix} 4 & 1 & 0 & 1 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & -3 & 4 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 3 & 1 \end{pmatrix}$$

$$= \begin{pmatrix} 4 & 1 & 0 & 1 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 5 \end{pmatrix}$$

Example

We calculate the new vector of right side of the equation:

$$b^{(3)} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ -2 \end{pmatrix} - b_3^{(2)} \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ -2 \end{pmatrix} - 7 \begin{pmatrix} 0 \\ 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ 5 \end{pmatrix}$$

Example

The vectors of multipliers are put in the matrix A under the diagonal:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 \end{pmatrix}$$

Example

Finally we have the set of equations with the upper triangular matrix:

$$\begin{pmatrix} 4 & 1 & 0 & 1 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 5 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ 5 \end{pmatrix}$$

Partial pivoting

If any $a_{kk}^{(k-1)}$ (pivot) is zero, k=1,2,...,n-1, the forward elimination process breaks down. Even if $a_{kk}^{(k-1)}$ is not identically zero, it may be so small in relation to the subsequent elements in the $k-th$ column that the multipliers generated are very large with a consequent amplification of round-off errors.

The partial pivoting has to be done.

Partial pivoting

1. We search, on and below the diagonal, the $k-th$ column of $A^{(k-1)}$ (the pivotal column) for the element of largest modulus. Suppose that it is $a_{rk}^{(k-1)}$ for some $r > k$.

2. If it is deemed to be close to 0, the matrix is "singular" and the elimination process is stopped.

3. Otherwise, for $r > k$ we interchange equation r and k and the elimination proceeds using the 'new' $k-th$ row.

Overview

- Introduction
- Gaussian elimination
  - Parallelization by vector operations
  - Blas implementation
- **Parallel backward substitution**

Triangular system

Backward substitiution

Solve the upper triangular system of equations:

$$A^{(n-1)}x = b^{(n-1)}$$

$A^{(n-1)}$

$$\begin{pmatrix}
a_{11} & \dots & a_{1,k-1} & a_{1,k} & \dots & a_{1,n-1} & a_{1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & a^{(k-2)}_{k-1,k-1} & a^{(k-2)}_{k-1,k} & \dots & a^{(k-2)}_{k-1,n-1} & a^{(k-2)}_{k-1,n} \\
0 & \dots & 0 & a^{(k-1)}_{k,k} & \dots & a^{(k-1)}_{k,n-1} & a^{(k-1)}_{k,n} \\
0 & \dots & 0 & 0 & \dots & a^{(k)}_{k+1,n-1} & a^{(k)}_{k+1,n} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots \\
0 & \dots & 0 & \dots & 0 & a^{(n-2)}_{n-1,n-1} & a^{(n-2)}_{n-1,n} \\
0 & \dots & 0 & 0 & \dots & 0 & a^{(n-1)}_{n,n}
\end{pmatrix}$$

Triangular system

$$x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$$

$$x_i = \frac{b_i^{(i-1)} - \sum_{j=i+1}^{n} a_{ij}^{(i-1)} x_j}{a_{ii}^{(i-1)}}$$

The code

```
 Void  backsubstitution(int n;float
a[][n],b[n],x[])
{
int  i,j; for (i=n-1;i>0;i-)
   { x[i]=b[i];
   for (j=i+1;j=n;j++)
     x[i]=x[i]-a[i,j]+x[j];
     x[i]=x[i]/a[i,i];
     };
}
```

Loop reordering

For parallel implementation we have to reorder the loop.

Example

$$\begin{pmatrix} a_{11} & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & a_{kk} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \dots & a_{n,n} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ \dots \\ x_k \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} a_{11}x_1 \\ \dots \\ a_{kk}x_k \\ \dots \\ a_{nn}x_n \end{pmatrix}$$

Example

$$\begin{pmatrix} a_{11}x_1 \\ a_{22}x_2 \\ \dots \\ a_{kk}x_k \\ \dots \\ a_{n-1,n-1}x_{n-1} \\ a_{nn}x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_k \\ \dots \\ b_{n-1} \\ b_n \end{pmatrix} - x_n \begin{pmatrix} a_{1,n} \\ a_{2,n} \\ \dots \\ a_{k,n} \\ \dots \\ a_{n-1,n} \\ 0 \end{pmatrix} - \dots - x_2 \begin{pmatrix} a_{12} \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix}$$

Example

Step 1
Calculate

$$x_n = \frac{b_n}{a_{nn}}$$

and decrease the size of the problem by 1.

Example

Step 2

We calculate the new right side vector:

$$\begin{pmatrix} b_1^{(1)} \\ b_2^{(1)} \\ \dots \\ b_k^{(1)} \\ \dots \\ b_{n-2}^{(1)} \\ b_{n-1}^{(1)} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_k \\ \dots \\ b_{n-2} \\ b_{n-1} \end{pmatrix} - x_n \begin{pmatrix} a_{1,} \\ a_{2,} \\ \dots \\ a_{k,n} \\ \dots \\ a_{n-2,n} \\ a_{n-1,n} \end{pmatrix}$$

Example

We have the new system of linear equations:

$$\begin{pmatrix} a_{11}x_1 \\ a_{22}x_2 \\ \dots \\ a_{kk}x_k \\ \dots \\ a_{n-2,n-2}x_{n-2} \\ a_{n-1,n-1}x_{n-1} \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(1)} \\ \dots \\ b_k^{(1)} \\ \dots \\ b_{n-2}^{(1)} \\ b_{n-1}^{(1)} \end{pmatrix} - x_{n-1} \begin{pmatrix} a_{1,n-1} \\ a_{2,n-1} \\ \dots \\ a_{k,n-1} \\ \dots \\ a_{n-2,n-1} \\ 0 \end{pmatrix} \dots - x_2 \begin{pmatrix} a_{12} \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix}$$

Project is co-financed by European Union within European Social Fund

Example

We can calculate x_{n-1}:

$$x_{n-1} = \frac{b_{n-1}^{(1)}}{a_{n-1,n-1}}$$

and the new right side vector

Example

Step 3

$$\begin{pmatrix} b_1^{(2)} \\ b_2^{(2)} \\ \dots \\ b_k^{(1)} \\ \dots \\ b_{n-3}^{(2)} \\ b_{n-2}^{(2)} \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(1)} \\ \dots \\ b_k^{(1)} \\ \dots \\ b_{n-3}^{(1)} \\ b_{n-2}^{(1)} \end{pmatrix} - x_{n-1} \begin{pmatrix} a_{1n-1} \\ a_{2n-1} \\ \dots \\ a_{kn-1} \\ \dots \\ a_{n-3,n-1} \\ a_{n-2,n-1} \end{pmatrix}$$

Example

The new set of equation has $n-2$ variables

$$\begin{pmatrix} a_{1,1}x_1 \\ a_{2,2}x_2 \\ \dots \\ a_{k,k}x_k \\ \dots \\ a_{n-3,n-3}x_{n-3} \\ a_{n-2,n-2}x_{n-2} \end{pmatrix} = \begin{pmatrix} b_1^{(2)} \\ b_2^{(2)} \\ \dots \\ b_k^{(1)} \\ \dots \\ b_{n-3}^{(2)} \\ b_{n-2}^{(2)} \end{pmatrix} - x_{n-3} \begin{pmatrix} a_{1,n-2} \\ a_{2,n-2} \\ \dots \\ a_{k,n-3} \\ \dots \\ a_{n-3,n-3} \\ 0 \end{pmatrix} \dots - x_2 \begin{pmatrix} a_{12} \\ 0 \\ \dots \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix}$$

$$x_{n-2} = \frac{b_{n-2}^{(2)}}{a_{n-2,n-2}}$$

Example

Step $(n-k)$

$$\begin{pmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{kk} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_k \end{pmatrix} = \begin{pmatrix} a_{11}x_1 \\ a_{22}x_2 \\ \dots \\ a_{kk}x_k \end{pmatrix}$$

Example

$$\begin{pmatrix} a_{11}x_1 \\ a_{22}x_2 \\ \dots \\ a_{k-1,k-1}x_{k-1} \\ a_{kk}x_k \end{pmatrix} = \begin{pmatrix} b_1^{(k-1)} \\ b_2^{(k-1)} \\ \dots \\ b_{k-1}^{(k-1)} \\ b_k^{(k-1)} \end{pmatrix} - x_k \begin{pmatrix} a_{1k-1} \\ a_{2k-1} \\ \dots \\ a_{kk-1} \\ 0 \end{pmatrix} - \dots - x_2 \begin{pmatrix} a_{12} \\ 0 \\ \dots \\ 0 \\ 0 \end{pmatrix}$$

$$x_k = \frac{b_k^{(k-1)}}{a_{k,k}}$$

Example

Step (n)

$$(a_{11})(x_1) = \left(b_1^{(n-1)}\right)$$

$$x_1 = \frac{b_1^{(n-1)}}{a_{11}}$$

Example

```
void par_backsubstitution
   (int n; float a[][n],b[n],x[])
{
   int i,j;
   for (i=n-1;i<0;i–)
        x[i]=b[i];
   for (j=i+1;j=n-1;j++)
        for (i=n-1;i<0;i–)
             x[i]=x[i]-a[i,j]+x[j];
   for (i=n-1;i<0;i–)
        x[i]=x[i]/a[i,i];
};
```

Example

The algorithm can be implemented using the BLAS2

Example

$$\begin{pmatrix} 4 & 1 & 0 & 1 \\ 0 & 3 & 2 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 5 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ 5 \end{pmatrix}$$

For parallel implementation we have to reorder the loop.

Example

$$\begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7 \\ 10 \\ 7 \\ 5 \end{pmatrix} - x_4 \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} - x_3 \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} - x_2 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Step 1

Calculate

$$x_4 = \frac{b_4}{a_{44}} = \frac{5}{5} = 1$$

Example

Step 2

New right side vector:

$$\begin{pmatrix} b_1^{(1)} \\ b_2^{(1)} \\ b_3^{(1)} \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} - x_4 \begin{pmatrix} a_{14} \\ a_{24} \\ a_{34} \end{pmatrix}$$

$$= \begin{pmatrix} 7 \\ 10 \\ 7 \end{pmatrix} - 1 \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 6 \\ 10 \\ 6 \end{pmatrix}$$

Example

We have the new set of equations with 3 variables:

$$\begin{pmatrix} 4 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 10 \\ 6 \end{pmatrix} - x_3 \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix} - x_2 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

We can calculate x_3:

$$x_3 = \frac{b_3^{(1)}}{a_{3,3}} = 2$$

Example

Step 3

$$\begin{pmatrix} b_1^{(2)} \\ b_2^{(2)} \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(1)} \end{pmatrix} - x_3 \begin{pmatrix} a_{13} \\ a_{23} \end{pmatrix} = \begin{pmatrix} 6 \\ 10 \end{pmatrix} - 2 \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} = \begin{pmatrix} 6 \\ 6 \end{pmatrix}$$

We have the new set of equations with 2 variables:

$$\begin{pmatrix} 4 & 0 \\ 0 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 6 \\ 6 \end{pmatrix} - x_2 \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$x_2 = \frac{b_2^{(2)}}{a_{2,2}} = \frac{6}{3} = 2$$

Example

Step 4

$$\left(b_1^{(3)} \right) = \left(b_1^{(2)} \right) - x_3 \left(a_{13} \right) = \left(6 \right) - 2 \left(1 \right) = 4$$

$$\left(4 \right) \left(x_1 \right) = \left(4 \right)$$

$$x_1 = 1$$

Thank you for your attention!

Any questions are welcome.