**Lecture Note-Numerical Analysis (6): Linear Algebraic Equations**

1. **Problem statement of the solution of linear algebraic equation**

**- calculate****satisfying** 



**- linear algebraic equation in the expanded form**



**(Example1) How to solve the following equation ?**

🡪

Solution

1. Eliminate the 1st element of the 2nd equation : (1행)X(-2) + (2행)



1. Eliminate the 1st element of the 3rd equation (1행)X(-2) + (3행)



1. Eliminate the 2nd element of the 3rd equation



1. Back substitution from the last equation to the first equation



The above solution method is the well-known Gauss Elimination method

1. **Familiarization to the matrix**

* **row vector** 
* **column vector** 
* **general rectangular matrix** 
* **square matrix** 
* **diagonal matrix** 
* **lower triangular matrix** 
* **upper triangular matrix** 
* **matrix transpose** 



**🡪** 

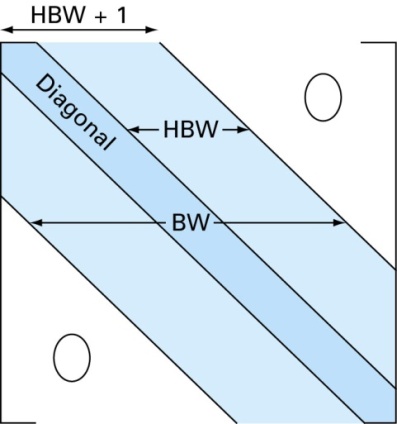
**Transpose of a row vector**

**🡪**  **becomes a column vector**

**Transpose of a column vector**

 **🡪**  **becomes a row vector**

* **banded matrix**



BW: Bandwidth(=2HBW+1)

HBW: Half-bandwidth



* **Identity matrix**



* **Definition of inverse matrix and its use in solving the linear algebraic equation.**

**- Inverse matrix of is defined as a matrix satisfying **

**- Its application**

** Therefore, we can calculate x if we know the matrix inverse** 

**(Question) Is the above method efficient? Answer: generally not efficient**

**(Question) How to calculate the matrix inverse?**

**Answer: by solving n-linear systems of algebraic equations**

****

**(example 2) Linear algebraic equation with the identity matrix: direct solution**



**(example 3) Linear algebraic equation with the diagonal matrix: direct substitution**



**(example 4) Linear algebraic equation with the upper triangular matrix: back substitution**

 :  can be computed

:  can be computed

:  and can be computed

**(example 5) Linear algebraic equation with the lower triangular matrix: forward substitution**

 :  is obtained

 :  is obtained

 :  is obtained

 :  is obtained

**From above examples, the simple substitution is enough to get the solution of the linear algebraic equation, when the leading matrix is one of the identity, diagonal, upper triangular, and lower triangular matrices.**

* **How to transform the general leading matrix  into one of the such forms?**

1. **Gauss elimination to solve a linear equation with small dimension**

* **Well-known traditional method as previously illustrated in (example 1) such as**

**(Example1) How to solve the following equation ?**

🡪

Solution

1. Eliminate the 1st element of the 2nd equation : (1행)X(-2) + (2행)



1. Eliminate the 1st element of the 3rd equation (1행)X(-2) + (3행)



1. Eliminate the 2nd element of the 3rd equation



**The leading matrix is reduced to the upper triangular matrix**

1. Back substitution from the last equation to the first equation



* **Final goal and Solution of triangular system**

**- Final goal: transform the full linear system of equation to an upper triangular system**



**where**





**- Solution of upper triangular system: Back Substitution**



1. the last equation:

 🡪

1. the 2nd from the last:

 🡪

1. the j-th equation (j=n-2,n-3,….., 1)



🡪

Operation Count

muliplication/divison = 1+2+….+(n-j+1)+….+(n+1)

addition/substraction =0+1+….+(n-j)+(n-1)

* **Naive Gauss elimination to successively remove the 1st column of the full sub-matrix**



If , the 1st equation can be written as



Multiply  to the above equation



Adding above equation to the j-th equation to eliminate -term as





Therefore, we can eliminate every term containing for. Furthermore, we can define a sub-matrix using the linear equation for and repeat the above procedure finally to get the upper triangular system. This upper triangular system can be easily solvable using the back substitution method, described above.

* **Gauss Elimination Method vs Gauss-Jordan method**

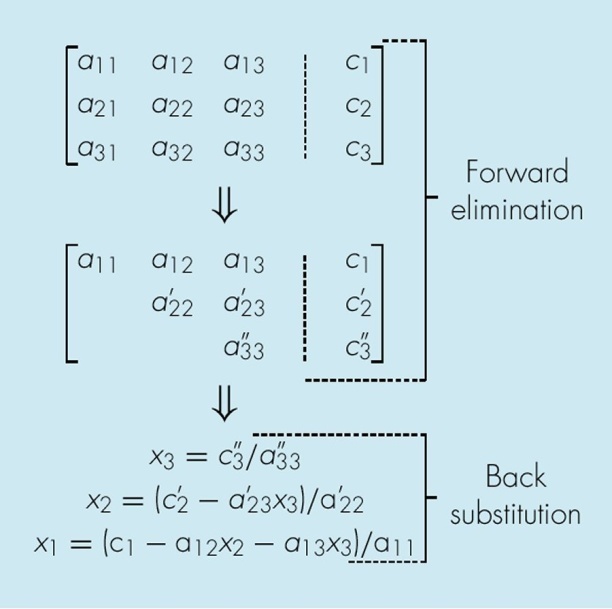
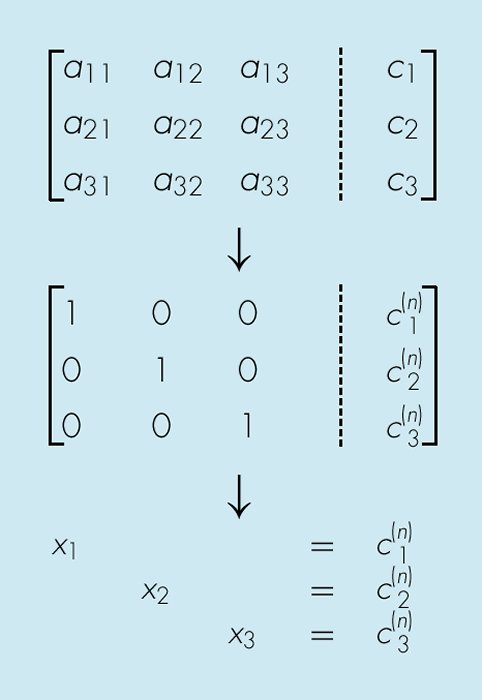
It is a variation of Gauss elimination. The major differences are:

* + When an unknown is eliminated, it is eliminated from all other equations rather than

just the subsequent ones.

* + All rows are normalized by dividing them by their pivot elements.
  + Elimination step results in an identity matrix.
  + Consequently, it is not necessary to employ back substitution to obtain solution.

Gauss Elimination Gauss-Jordan method

* **Pitfalls of Gauss Elimination Methods**
* **Division by zero**. It is possible that during both elimination and back-substitution phases a division by zero can occur.
* **Large round-off errors probable.**
* **Ill-conditioned systems**. Systems where small changes in coefficients result in large changes in the solution. Alternatively, it happens when two or more equations are nearly identical, resulting a wide ranges of answers to approximately satisfy the equations. Since round off errors can induce small changes in the coefficients, these changes can lead to large solution errors.
* **Singular systems**. When two equations are identical, we would loose one degree of freedom and be dealing with the impossible case of n-1 equations for n unknowns. For large sets of equations, it may not be obvious however. The fact that the determinant of a singular system is zero can be used and tested by computer algorithm after the elimination stage. If a zero diagonal element is created, calculation is terminated.
* **Techniques for Improving Solutions**
* **Use more significant figures (double rather than float at sacrifice of memory/CPU time)**
* **Scaling**
* **Pivoting**. If a pivot element is zero, normalization step leads to division by zero. The same problem may arise, when the pivot element is close to zero. Problem can be avoided:
  + **Partial pivoting**. Switching the rows so that the largest element is the pivot element.
  + **Full(complete) pivoting**. Searching for the largest element in all rows and columns then switching.

1. **Pivoted Gauss Elimination**

* **Pivoting Examples for** 

🡪

(Step 1)

🡪🡪 with 

🡪

(Step 2)

 with 

(Step 3)

 with 

* **Partial Pivoting Strategy for the Sub matrix**
  + Find the row number(Imax) which has the maximum absolute value in the first column
  + Place this row to the first row and the original first row to (Imax)-th row
* **Full Pivoting Strategy**
  + Find the element number (Imax,Jmax) which has the maximum absolute value in the full sub-matrix
  + Use the elementary row-operation and the elementary column-oparation
  + After the solution is obtained, the solution corresponding to the original equation should be recovered.
* **Effect of Elementary Row Operation for both**  **and** 







We can get the same solution after the row operation for both  and 

* **Effect of Elementary Column Operation only for**

The order of the solution is exchanged between j-th column and k-th column.

* **How to cope with several column operation?**

**Answer: use the order definition vector “INDEX”**

1. **If we use “INDEX” to define the solution order, set the initial value of INDEX**

**!-------------------------------------------------------------**

**For i=1:n**

**INDEX(i) = i**

**End**

**!-------------------------------------------------------------**

1. **Change INDEX whenever the column operation is occurred between j-th column and k-th column**

**!-------------------------------------------------------------**

**INDEX\_tmp = INDEX(j)**

**INDEX(j) = INDEX(k)**

**INDEX(k) = INDEX\_tmp**

**!-------------------------------------------------------------**

1. **After solving the linear algebraic equation, recover the solution order as following**

**!------ x\_bar : final solution of**  **------------**

**!------ x : original solution of**  **------------**

**For j=1:n**

**x(j) = x\_bar(INDEX(j))**

**End**

**[Appendix A] Introduction to Permutation Matrix (Related to Pivoting)**

**A-1. Example for 5x5 square matrix**

**, **

**(A-1-1) Switching 2nd and 3th rows of I**

**,**

** provides the row permutation between the 2nd and 3rd rows**

****

** provides the column permutation between the 2nd and 3rd columns**

**(A-1-2) Switching 2nd and 4th rows of I**

**, **

** provides the row permutation between the 2nd and 4th  rows**

** provides the column permutation between the 2nd and 4th columns**

**A-2. In general Matrix , **

**Switching i-th and j-th row of I causes**

** provides the row permutation between the i-th and j-th rows 🡪 the left permutation**

** provides the column permutation between the i-th and j-th columns🡪 the right permutation**

****

**[Appendix B] Matrix form of Gauss-Jordan Elimination Method**

**B-1. Example for the first column operation for** 

**, **

** 🡪**

**🡪**

With the same procedure for the sub-block square matrix, we can remove the  Using

** **

Finally, we reach to the equation with a diagonal matrix  as

**** 🡪 **** where **** and ****

**[Appendix C] Algorithm for the upper and lower triangular system**

* + - **Solution of upper triangular system: Back Substitution**



1. the last equation:

 🡪

1. the 2nd from the last:

 🡪

1. the j-th equation (j=n-2,n-3,….., 1)



🡪

* **Solution of lower triangular system: Forward Substitution**



1. the 1st equation:  🡪
2. the j-th equation (j=2,3,4,….,n)



🡪