

LU Decomposition

1.0 Introduction

We have seen how to construct the Y-bus used in the matrix equation

$$\underline{I} = \underline{Y}\underline{V} \quad (1)$$

If we are given the bus voltages, we can construct \underline{Y} and then very easily find \underline{I} . Unfortunately, this is not generally what we know. Rather, we typically know \underline{I} (because we know generation and load), and then we must find \underline{V} . This is easy enough, using matrix inverses, i.e.,

$$\underline{V} = \underline{Y}^{-1} \underline{I} \quad (2)$$

However, there is a very practical problem with this. A model of the eastern US interconnection can be 50000 buses. This means that the Y-bus for this model is a 50000×50000 matrix. Direct matrix inversion for this dimensionality is computational suicide.

Fortunately, there is an alternative to direct matrix inversion. We will never actually get the inverse, but we will solve for \underline{V} given \underline{I} in eq. (1).

The method that allows us to do this is called LU decomposition. It is actually a very widely known and used method in many different disciplines.

In fact, using it to solve eq. (1) is not the most common application in power systems. A much more common application of LU decomposition is in the numerical, iterative algorithm used to solve the power flow problem. But we will come to this later. For now, let's learn LU-decomposition on the generic problem $\underline{A} \underline{x} = \underline{b}$, motivated by the specific application $\underline{Y} \underline{V} = \underline{I}$.

If you have taken a linear algebra course (from Math), this is familiar to you. If you have not taken such a course, you should.

2.0 Forward-backward substitution

Consider that you are able to obtain \underline{A} (or \underline{Y}) as the product of two special matrices, i.e.,

$$\underline{A} = \underline{L}\underline{U} \quad (3)$$

that satisfy the following:

- Both \underline{L} and \underline{U} are square matrices of the same dimension as \underline{A} .
- \underline{U} is an upper-triangular matrix, meaning that all elements below the diagonal are 0.
- \underline{L} is a lower-triangular matrix, meaning that all elements above the diagonal are 0.
- All diagonal elements of \underline{U} are 1.

So, for a 3×3 case, we would have:

$$\begin{aligned} \underline{A} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \\ &= \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix} = \underline{L}\underline{U} \quad (4) \end{aligned}$$

For the moment, let's not worry about how to obtain \underline{L} and \underline{U} . Rather, let's think about what we can do if we get them.

What we know at this point is that $\underline{A} = \underline{L}\underline{U}$, and $\underline{A}\underline{x} = \underline{b}$. Therefore,

$$\underline{L}\underline{U}\underline{x} = \underline{b} \quad (5)$$

Define:

$$\underline{w} = \underline{U}\underline{x} \quad (6)$$

Substitution of (6) into (5) results in

$$\underline{L}\underline{w} = \underline{b} \quad (7)$$

The situation is the following. We want to find \underline{x} . It appears that (6) and (7) are not very helpful, because solving them for \underline{x} and \underline{w} , respectively, will require an inverse. But let's take a closer look at eqs. (6) and (7), in terms of the fully expressed matrix relations and see if we can get \underline{x} and \underline{w} without matrix inversion. If so, then our procedure will be to use (7) to find \underline{w} and then (6) to find \underline{x} .

$$\underline{\mathbf{w}} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} \mathbf{1} & u_{12} & u_{13} \\ \mathbf{0} & \mathbf{1} & u_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \underline{\mathbf{U}} \underline{\mathbf{x}} \quad (8)$$

$$\underline{\mathbf{L}} \underline{\mathbf{w}} = \begin{bmatrix} l_{11} & \mathbf{0} & \mathbf{0} \\ l_{21} & l_{22} & \mathbf{0} \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \underline{\mathbf{b}} \quad (9)$$

Observe that

- Equation (9) can be solved for $\underline{\mathbf{w}}$ without inverting $\underline{\mathbf{L}}$ and
- Equation (8) could then be solved for $\underline{\mathbf{x}}$ without inverting $\underline{\mathbf{U}}$.

Let's start with (9). We see that we can begin with the row 1 equation and proceed as follows:

$$l_{11}w_1 = b_1 \Rightarrow w_1 = b_1 / l_{11} \quad (10)$$

$$l_{21}w_1 + l_{22}w_2 = b_2 \Rightarrow w_2 = \frac{b_2 - l_{21}w_1}{l_{22}} \quad (11)$$

$$l_{31}w_1 + l_{32}w_2 + l_{33}w_3 = b_3 \Rightarrow w_3 = \frac{b_3 - l_{31}w_1 - l_{32}w_2}{l_{33}} \quad (12)$$

This procedure is called **forward substitution**. Consideration of the pattern of calculation introduced by eqs. (10)-(12) suggests a generalized formula for forward substitution, useful for computer programming, as follows:

$$w_k = \frac{b_k - \sum_{j=1}^{k-1} l_{kj} w_j}{l_{kk}} \quad (13)$$

Now that we have \underline{w} , we can use (8) to find \underline{x} . We see that we can begin with the row 3 equation and proceed as follows:

$$x_3 = w_3 \quad (14)$$

$$x_2 + u_{23}x_3 = w_2 \Rightarrow x_2 = w_2 - u_{23}x_3 \quad (15)$$

$$x_1 + u_{12}x_2 + u_{13}x_3 = w_1 \Rightarrow x_1 = w_1 - u_{12}x_2 - u_{13}x_3 \quad (16)$$

This procedure is called **backward substitution**. Consideration of the pattern of calculation introduced by eqs. (14)-(16) suggests a generalized formula for forward substitution, useful for computer programming, as follows:

$$\mathbf{x}_k = \mathbf{w}_k - \sum_{j=k+1}^n \mathbf{u}_{kj} \mathbf{x}_j \quad (17)$$

where n is the dimension of the matrix.

3.0 Factorization using Crout algorithm

So we see that if we have $\underline{\mathbf{L}}$ and $\underline{\mathbf{U}}$, we can solve $\underline{\mathbf{A}} \underline{\mathbf{x}} = \underline{\mathbf{b}}$ for $\underline{\mathbf{x}}$. So natural question at this point is: How to find $\underline{\mathbf{L}}$ and $\underline{\mathbf{U}}$?

The method of finding $\underline{\mathbf{L}}$ and $\underline{\mathbf{U}}$ from $\underline{\mathbf{A}}$ is called the LU factorization of $\underline{\mathbf{A}}$, otherwise known as the LU-decomposition of $\underline{\mathbf{A}}$. You will enjoy factorization 😊.

To motivate it, let's first look back at eq. (4).

$$\begin{aligned}
\underline{\mathbf{A}} &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \\
&= \begin{bmatrix} l_{11} & \mathbf{0} & \mathbf{0} \\ l_{21} & l_{22} & \mathbf{0} \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} \mathbf{1} & u_{12} & u_{13} \\ \mathbf{0} & \mathbf{1} & u_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \end{bmatrix} = \underline{\mathbf{L}}\underline{\mathbf{U}} \quad (4)
\end{aligned}$$

Based on eq. (4), we see that the following sequence of calculations may be performed.

- Row 1 of $\underline{\mathbf{A}}$ and $\underline{\mathbf{L}}$, across columns of $\underline{\mathbf{U}}$.

$$a_{11} = l_{11}$$

$$a_{12} = l_{11}u_{12} \Rightarrow u_{12} = a_{12} / l_{11}$$

$$a_{13} = l_{11}u_{13} \Rightarrow u_{13} = a_{13} / l_{11}$$

- Row 2 of $\underline{\mathbf{A}}$ and $\underline{\mathbf{L}}$, across columns of $\underline{\mathbf{U}}$.

$$a_{21} = l_{21}$$

$$a_{22} = l_{21}u_{12} + l_{22} \Rightarrow l_{22} = a_{22} - l_{21}u_{12}$$

$$a_{23} = l_{21}u_{13} + l_{22}u_{23} \Rightarrow u_{23} = \frac{a_{23} - l_{21}u_{13}}{l_{22}}$$

- Row 3 of $\underline{\mathbf{A}}$ and $\underline{\mathbf{L}}$, across columns of $\underline{\mathbf{U}}$:

$$a_{31} = l_{31}$$

$$a_{32} = l_{31}u_{12} + l_{32} \Rightarrow l_{32} = a_{32} - l_{31}u_{12}$$

$$a_{33} = l_{31}u_{13} + l_{32}u_{23} + l_{33} \Rightarrow l_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

The above is convincing evidence that we will be able to perform the desired factorization. Although we have done so for only a 3×3 case, it is easy to see that the procedure would also work for a matrix of any dimension.

If you study closely the pattern of calculation, you can convince yourself that the following generalized formula can be used in computer programming [1].

$$l_{ij} = a_{ij} - \sum_{s=1}^{j-1} l_{is} u_{sj} \quad (18)$$

$$u_{ij} = \frac{a_{ij} - \sum_{s=1}^{i-1} l_{is} u_{sj}}{l_{ii}} \quad (19)$$

Use of eqs. (18,19) comprise what is known as the Crout algorithm.

4.0 Method of Bergen & Vittal (Dolittle)

$$\underline{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} = \underline{LU} \quad (4a)$$

- Row 1 of \underline{A} and \underline{L} , across columns of \underline{U} .

$$a_{11} = u_{11}$$

$$a_{12} = u_{12}$$

$$a_{13} = u_{13}$$

- Row 2 of \underline{A} and \underline{L} , across columns of \underline{U} .

$$a_{21} = l_{21}u_{11} \Rightarrow l_{21} = \frac{a_{21}}{u_{11}}$$

$$a_{22} = l_{21}u_{12} + u_{22} \Rightarrow u_{22} = a_{22} - l_{21}u_{12}$$

$$a_{23} = l_{21}u_{13} + u_{23} \Rightarrow u_{23} = a_{23} - l_{21}u_{13}$$

- Row 3 of \underline{A} and \underline{L} , across columns of \underline{U} :

$$a_{31} = l_{31}u_{11} \Rightarrow l_{31} = \frac{a_{31}}{u_{11}}$$

$$a_{32} = l_{31}u_{12} + l_{32}u_{22} \Rightarrow l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}}$$

$$a_{33} = l_{31}u_{13} + l_{32}u_{23} + u_{33} \Rightarrow u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

5.0 Factorization using Gaussian elimination

Trick: Augment A matrix before you begin the below algorithm by adding the vector b as the n+1 column. Then, when you finish the algorithm, you will have the vector w in the n+1 column.

The algorithm is as follows:

1. Perform Gaussian elimination on \underline{A} . Let $i=1$. In each repetition below, row i is the *pivot row* and a_{ii} is the *pivot*.
 - a. $L_{ji}=a_{ji}$ for $j=i, \dots, n$.
 - b. Divide row i by a_{ii} .
 - c. If $[i=n, \text{ go to 2}]$ else $[\text{go to d}]$.
 - d. Eliminate all a_{ji} , $j=i+1, \dots, n$. This means to make all elements directly

beneath the pivot equal to 0 by adding an appropriate multiple of the pivot row to each row beneath the pivot.

e. $i=i+1$, go to a.

2. The matrix U is what remains.

Example: Use LU decomposition to solve for x in the below system of equations.

$$3x_1 + 3x_2 + 6x_3 + 9x_4 = 1$$

$$x_1 + 3x_2 + 4x_3 + x_4 = 3$$

$$x_1 + 2x_2 + 5x_3 + 6x_4 = 2$$

$$x_1 - x_3 + 3x_4 = 1$$

In matrix form, the above is:

$$\begin{bmatrix} 3 & 3 & 6 & 9 \\ 1 & 3 & 4 & 1 \\ 1 & 2 & 5 & 6 \\ 1 & 0 & -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ 2 \\ 1 \end{bmatrix}$$

Form the augmented matrix.

$$\begin{bmatrix} 3 & 3 & 6 & 9 & 1 \\ 1 & 3 & 4 & 1 & 3 \\ 1 & 2 & 5 & 6 & 2 \\ 1 & 0 & -1 & 3 & 1 \end{bmatrix}$$

Now perform the algorithm.

$$\begin{bmatrix} 3 & 3 & 6 & 9 & 1 \\ 1 & 3 & 4 & 1 & 3 \\ 1 & 2 & 5 & 6 & 2 \\ 1 & 0 & -1 & 3 & 1 \end{bmatrix}; L = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 1 & - & 0 & 0 \\ 1 & - & - & 0 \\ 1 & - & - & - \end{bmatrix}$$

Divide first row by 3 and then add multiples of it to remaining rows so that first element in remaining rows gets zeroed.

$$\begin{bmatrix} 1 & 1 & 2 & 3 & 1/3 \\ 0 & 2 & 2 & -2 & 8/3 \\ 0 & 1 & 3 & 3 & 5/3 \\ 0 & -1 & -3 & 0 & 2/3 \end{bmatrix}; L = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 1 & - & 0 \\ 1 & -1 & - & - \end{bmatrix}$$

Divide second row by 2 and then add multiples of it to remaining rows so that second element in remaining rows gets zeroed.

$$\begin{bmatrix} 1 & 1 & 2 & 3 & 1/3 \\ 0 & 1 & 1 & -1 & 4/3 \\ 0 & 0 & 2 & 4 & 1/3 \\ 0 & 0 & -2 & -1 & 6/3 \end{bmatrix}; \quad L = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 1 & 2 & 0 \\ 1 & -1 & -2 & - \end{bmatrix}$$

Divide third row by 2 and then add multiples of it to remaining rows so that third element in remaining rows gets zeroed.

$$\begin{bmatrix} 1 & 1 & 2 & 3 & 1/3 \\ 0 & 1 & 1 & -1 & 4/3 \\ 0 & 0 & 1 & 2 & 1/6 \\ 0 & 0 & 0 & 3 & 7/3 \end{bmatrix}; \quad L = \begin{bmatrix} 3 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 1 & 2 & 0 \\ 1 & -1 & -2 & 3 \end{bmatrix}$$

Divide third row by 3.

$$\begin{bmatrix} 1 & 1 & 2 & 3 & 1/3 \\ 0 & 1 & 1 & -1 & 4/3 \\ 0 & 0 & 1 & 2 & 1/6 \\ 0 & 0 & 0 & 1 & 7/9 \end{bmatrix}$$

[1] Vlach and Singhal, "Computer Methods for Circuit Analysis and Design," 2nd edition, 1994.