International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 6 Issue 8, August - 2022, Pages: 58-64

Jacobi Method for Solving Linear System of Equations

Safaa M. Aljassas[1] and Ahmed Sabah Al-Jilawi[2]

[1]Iraq\ University of Kufa \ College of Education for girls \ Mathematics Dep.
safaam.musa@uokufa.edu.iq
[2]Iraq\ University of Babylon\ Faculty of Basic Education\ Mathematics Dep.
Aljelawy2000@yahoo.com

Abstract: *The main goal of this research to solve systems of linear equations using an iterative method, Jacobi method, and using an algorithm in Matlab language.*

Keywords: Jacobi method, iterative method, linear system of equations.

1. Introduction.

Systems of linear equations arise in a large number of areas, both directly in modeling physical situations and indirectly in the numerical solution of other mathematical models. These applications occur in virtually all areas of the physical, biological, and social sciences. In addition, linear systems are involved in the following: optimization theory; solving systems of nonlinear equations; the approximation of functions; the numerical solution of boundary value problems for ordinary differential equations, partial differential equations, and integral equations; statistical inference; and numerous other problems. Because of the widespread importance of linear systems, much research has been devoted to their numerical solution. Excellent algorithms have been developed for the most common types of problems for linear systems, and some of these are defined, analyzed, and illustrated in this chapter. The most common type of problem is to solve a square linear system> of moderate order, with coefficients that are mostly nonzero. Such linear systems, of any order, are called dense. For such systems, the coefficient matrix A must generally be stored in the main memory of the computer in order to efficiently solve the linear system, and thus memory storage limitations in most computers will limit the order of the system. With the rapid decrease in the cost of computer memory, quite large linear systems can be accommodated on some machines, but it is expected that for most smaller machines, the practical upper limits on the order will be of size 100 to 500. Most algorithms for solving such dense systems are based on Gaussian elimination. It is a direct method in the theoretical sense that if rounding errors are ignored, then the exact answer is found in a finite number of· steps. Modifications for improved error behavior with Gaussian elimination, variants for special classes of matrices.

2. linear System of Equations [7]

We can write the general system that content m equations and n variables as follow:-

$$\left.\begin{aligned} &a_{11}x_1 + a_{12}x_2 + \cdots\cdots + a_{1n}x_n = b_1 \\ &a_{21}x_1 + a_{22}x_2 + \cdots\cdots + a_{2n}x_n = b_2 \\ &\vdots \\ &a_{m1}x_1 + a_{m2}x_2 + \cdots\cdots + a_{mn}x_n = b_m \end{aligned}\right\} \quad \cdots(1)$$

By using matrices

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots\cdots & a_{1n} \\ a_{21} & a_{22} & \cdots\cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots\cdots & a_{mn} \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

Now, we can rewrite (1) as follows :- $Ax = b$

3. Jacobi Method [7]

the Jacobi method (or Jacobi iterative method) is an algorithm for determining the solutions of a diagonally dominant system of linear equations. Each diagonal element is solved for, and an approximate value is plugged in. The process is then iterated until it converges. This algorithm is a stripped-down version of the Jacobi transformation method of matrix diagonalization. The method is named after Carl Gustav Jacob Jacobi.

From the linear system of equations:

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$
$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2$$
$$\vdots$$
$$a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n$$

We generate the iterative formats:

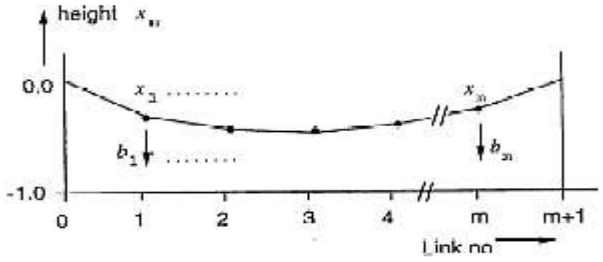
$$x_1^{(k+1)} = \frac{b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)}}{a_{11}}$$
$$x_2^{(k+1)} = \frac{b_2 - a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)}}{a_{22}}$$
$$\vdots$$
$$x_n^{(k+1)} = \frac{b_n - a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - \cdots - a_{n,n-1}x_{n-1}^{(k)}}{a_{nn}}$$, Where k=0, 1,….

Stop condition is $\left|x_i^{(k+1)} - x_i^{(k)}\right| < \in$ for any k and all i=1, 2, …,n.

To solve above system, we need an approximate solution $x^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \vdots \\ x_n^{(0)} \end{bmatrix}$

In general $x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij}x_j^{(k)} \right)$, $i = 1,2,...,n$, $k = 1,2,....$, $a_{ii} \neq 0 \ \forall i = 1,2,...,n$

4. **Example**: Jacobi solution of weighted chain [8].



Consider a hanging chain of $m + 1$ light links with fixed ends at height $x_0 = x_{m+1} = 0.$
It could be the supporting chain for the Clifton suspension bridge in Bristol, shown $\Rightarrow$
There is unit tension in the chain, the links are of unit Length. a downward force b_j is applied on the jth hinge. If we make the approximation that the links make small angles to the horizontal, resolving forces vertically at each hinge gives:

$$x_{j-1} - 2x_j + x_{j+1} = b_j \qquad ,1 \leq j \leq m$$

This is a matrix equation of the form Ax = b . In the case m = 5, there are 5 links

$$A = \begin{bmatrix} -2 & 1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 1 & -2 \end{bmatrix}$$

Dividing the matrix A into D + L+ U , and taking the case where $b = [1,1,1,1,1]^T$ gives the Jacobi iteration algorithm in the form $x_{n+1} = -D^{-1}(L + U)x_n + D^{-1} b = Bx_n + c$:-

$$x_{n+1} = \begin{bmatrix} 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 \end{bmatrix} x_n - \begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{bmatrix}$$

5. **Definition [2]:-** A matrix $A_{n\times n}$ is said to be diagonally dominant iff, for each $i = 1,2,...,n$ $|a_{ii}| > \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}| < 1$

6. **Theorem [2]** The convergence condition (for any iterative method) is when the matrix A is diagonally dominant.

Note:- Condition of convergence in Jacobi Method is $\max \sum_{\substack{j=1 \\ j\neq i}}^{n} \left|\frac{a_{ij}}{a_{ii}}\right| < 1$, $i = 1,2,...,n$

7. Examples and results

Example (1):-

Use the Jacobi method to approximate the solution of the following system of linear equations.

$5x_1 - 2x_2 + 3x_3 = 12$

$-3x_1 + 9x_2 + x_3 = 14$

$2x_1 - x_2 - 7x_3 = -12$

Solution To begin, write the system in the form

$x_1^{k+1} = (1/5)\left(12 + 2x_2^k - 3x_3^k\right)$

$x_2^{k+1} = (1/9)\left(14 + 3x_1^k - x_3^k\right)$

$x_3^{k+1} = (-1/7)\left(-12 - 2x_1^k + x_2^k\right)$

Where k=0, 1,…. , Initial approximation $x_1 = 0$, $x_2 = 0$, $x_3 = 0$

When applying the Jacobi method to the above-mentioned system, we obtained the results shown in Table (1)

Table (1)

k	x_1	x_2	x_3
0	0.000000	0.000000	0.000000
1	2.400000	1.555556	1.714286
2	1.993651	2.165079	2.177778
3	1.959365	1.978131	1.974603
4	2.006490	1.989277	1.991514
5	2.000802	2.003106	2.003386
6	1.999211	1.999891	1.999785
7	2.000085	1.999761	1.999790
8	2.000030	2.000052	2.000059
9	1.999986	2.000004	2.000001
10	2.000001	1.999995	1.999995
11	2.000001	2.000001	2.000001
12	2	2	2
13	2	2	2

We note that the iterations in the three columns are identical, so $x_1 = 2$, $x_2 = 2$, $x_3 = 2$ when k=12 and k=13.

Example (2):-

Use the Jacobi method to approximate the solution of the following system of linear equations.

$$10x_1 - x_2 + 2x_3 = 6$$
$$-x_1 + 11x_2 - x_3 + 3x_4 = 25$$
$$2x_1 - x_2 + 10x_3 - x_4 = -11$$
$$3x_2 - x_3 + 8x_4 = 15$$

Solution To begin, write the system in the form

$$x_1^{k+1} = (0.1)\left(6 + x_2^k - 2x_3^k\right)$$
$$x_2^{k+1} = (1/11)\left(25 + x_1^k + x_3^k - 3x_4^k\right)$$
$$x_3^{k+1} = (0.1)\left(-11 - 2x_1^k + x_2^k + x_4^k\right)$$
$$x_4^{k+1} = (1/8)\left(15 - 3x_2^k + x_3^k\right)$$

Where k=0, 1,…. , Initial approximation $x_1 = 0$,$x_2 = 0$,$x_3 = 0$,when applying the Jacobi method to the above-mentioned system, we obtained the results shown in Table (2).

We note that the iterations of values x_1, x_3 are identical when k=17 and k=18 while the iterations of values x_2 ,x_4 are identical when k=18 and k=19 , so $x_1 = 1$, $x_2 = 2$, $x_3 = -1$, $x_4 = 1$.

Table (2)				
k	**x_1**	**x_2**	**x_3**	**x_4**
0	0	0	0	0
1	0.600000	2.272727	-1.100000	1.875000
2	1.047273	1.715909	-0.805227	0.885227
3	0.932636	2.053306	-1.049341	1.130881
4	1.015199	1.953696	-0.968109	0.973843
5	0.988991	2.011415	-1.010286	1.021351
6	1.003199	1.992241	-0.994522	0.994434
7	0.998128	2.002307	-1.001972	1.003594
8	1.000625	1.998670	-0.999036	0.998888
9	0.999674	2.000448	-1.000369	1.000619
10	1.000119	1.999768	-0.999828	0.999786
11	0.999942	2.000085	-1.000068	1.000109
12	1.000022	1.999959	-0.999969	0.999960
13	0.999990	2.000016	-1.000013	1.000019
14	1.000004	1.999993	-0.999994	0.999992
15	0.999998	2.000003	-1.000002	1.000003
16	1.000001	1.999999	-0.999999	0.999999
17	1	2.000001	-1.	1.000001
18	1	2	-1	1
19	1	2	-1	1

Example (3):-
Use the Jacobi method to approximate the solution of the following system of linear equations.

$$4x_1 + x_2 - x_3 = 3$$
$$x_1 + 6x_2 - 2x_3 + x_4 - x_5 = -6$$
$$x_2 + 5x_3 - x_5 + x_6 = -5$$
$$2x_2 + 5x_4 - x_5 - x_7 - x_8 = 0$$
$$-x_3 - x_4 + 6x_5 - x_6 - x_8 = 12$$
$$-x_3 - x_5 + 5x_6 = -12$$
$$-x_4 + 4x_7 - x_8 = -2$$
$$-x_4 - x_5 - x_7 + 5x_8 = 2$$

Solution To begin, write the system in the form

$$x_1^{k+1} = (0.25)\left(3 - x_2^k + x_3^k\right)$$

$$x_2^{k+1} = (1/6)\left(-6 - x_1^k + 2x_3^k - x_4^k + x_5^k\right)$$

$$x_3^{k+1} = (1/5)\left(-5 - x_2^k + x_5^k - x_6^k\right)$$

$$x_4^{k+1} = (1/5)\left(-2x_2^k + x_5^k + x_7^k + x_8^k\right)$$

$$x_5^{k+1} = (1/6)\left(12 + x_3^k + x_4^k + x_6^k + x_8^k\right)$$

$$x_6^{k+1} = (1/5)\left(-12 + x_3^k + x_5^k\right)$$

$$x_7^{k+1} = (0.24)\left(-2 + x_4^k + x_8^k\right)$$

$$x_8^{k+1} = (1/5)\left(2 + x_4^k + x_5^k + x_7^k\right)$$

Where k=0, 1,…. , Initial approximation $x_1 = 0.1$, $x_2 = 0.1$, $x_3 = 0.1$, $x_4 = 0.1$, $x_5 = 0.1$, $x_6 = 0.1$, $x_7 = 0.1$,$x_8 = 0.1$ when applying the Jacobi method to the above-mentioned system, we obtained the results shown in Table (3).

Table (3)

k	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
1	0.750000	-0.983333	-1.020000	0.020000	2.066667	-2.360000	-0.450000	0.460000
2	0.740833	-1.123889	0.082000	0.808667	1.516667	-2.190667	-0.380000	0.727333
3	1.051472	-0.978139	-0.033756	0.822356	1.904556	-2.080267	-0.116000	0.789067
4	0.986096	-1.006131	-0.007408	0.906780	1.916233	-2.025840	-0.097144	0.922182
5	0.999681	-0.998576	-0.010359	0.950706	1.965952	-2.018235	-0.042759	0.945174
6	0.997054	-1.000859	-0.003447	0.973104	1.977881	-2.008881	-0.026030	0.974780
7	0.999353	-0.999862	-0.002476	0.985670	1.989259	-2.005113	-0.013029	0.984991
8	0.999347	-1.000119	-0.001153	0.992189	1.993845	-2.002643	-0.007335	0.992380
9	0.999742	-0.999999	-0.000678	0.995826	1.996795	-2.001462	-0.003858	0.995740
10	0.999830	-1.000021	-0.000349	0.997735	1.998238	-2.000777	-0.002109	0.997753
11	0.999918	-1.000004	-0.000193	0.998785	1.999060	-2.000422	-0.001128	0.998773
12	0.999953	-1.000005	-0.000103	0.999343	1.999490	-2.000226	-0.000611	0.999343
13	0.999976	-1.000002	-0.000056	0.999647	1.999726	-2.000122	-0.000328	0.999645
14	0.999987	-1.000001	-0.000030	0.999809	1.999852	-2.000066	-0.000177	0.999809
15	0.999993	-1.000001	-0.000016	0.999897	1.999920	-2.000036	-0.000095	0.999897
16	0.999996	-1	-0.000008	0.999945	1.999957	-2.000019	-0.000051	0.999944
17	0.999998	-1	-0.000003	0.999970	1.999977	-2.000010	-0.000028	0.999970
18	0.999999	-1	-0.000001	0.999984	1.999988	-2.000006	-0.000015	0.999984
19	0.999999	-1	-0.000001	0.999991	1.999993	-2.000003	-0.000008	0.999991
20	1	-1	-0.000001	0.999996	1.999997	-2.000002	-0.000005	0.999996
21	1	-1	0	0.999999	1.999999	-2.000001	-0.000001	0.999999
22	1	-1	0	1	2	-2	0	1
23	1	-1	0	1	2	-2	0	1

We note that the iterations of value x_1 is identical when k=20 and k=21 , while the iterations of value x_2 is identical when k=16 and k=17, x_3 is identical when k=21 and k=22 while they were the iterations of values x_4 , x_5 , x_6, x_7 ,x_8, are identical when k=22 and k=23 so $x_1 = 1$, $x_2 = -1$, $x_3 = 0$, $x_4 = 1$, $x_5 = 2$, $x_6 = -2$, $x_7 = 0$,$x_8 = 1$.

8. Conclusion

Jacobian method or **Jacobi method** is one the iterative methods for approximating the solution of a system of n linear equations in n variables. The Jacobi iterative method is considered as an iterative algorithm which is used for determining the solutions for the system of linear equations in numerical linear algebra, which is diagonally dominant. In this method, an approximate value is filled in for each diagonal element. Until it converges, the process is iterated. This algorithm was first called the Jacobi transformation process of matrix diagonalization. Jacobi Method is also known as the simultaneous displacement method.

References

1. Acton, F. S. Numerical Methods That Work, 2nd printing. Washington, DC: Math. Assoc. Amer., pp. 161-163, 1990.
2. Adel Rashid,Muhannad Nafeh, Numerical Analysis Lectures,Dean of the College of Mathematics, College of Education for Pure Sciences, Ibn Al-Haitham University, Baghdad, 2018-2019
3. Barrett, R.; Berry, M.; Chan, T. F.; Demmel, J.; Donato, J.; Dongarra, J.; Eijkhout, V.; Pozo, R.; Romine, C.; and van der Vorst, H. Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd ed. Philadelphia, PA: SIAM, 1994. http://www.netlib.org/linalg/html_templates/Templates.html.
4. Bronshtein, I. N. and Semendyayev, K. A. Handbook of Mathematics, 3rd ed. New York: Springer-Verlag, p. 892, 1997.
5. Hageman, L. and Young, D. Applied Iterative Methods. New York: Academic Press, 1981.
6. Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. Numerical Recipes in FORTRAN: The Art of Scientific Computing, 2nd ed. Cambridge, England: Cambridge University Press, pp. 864-866, 1992.
7. Saeed, Rostom Karim, Numerical Analysis Course Outline, Salahaddin University / Erbil, College of Science - Mathematics Department, 2007
8. Stephen Roberts, Iterative Solution of Simultaneous Equations, ENGINEERING COMPUTATION, Lecture 3
9. Varga, R. Matrix Iterative Analysis. Englewood Cliffs, NJ: Prentice-Hall, 1962.
10. Young, D. Iterative Solutions of Large Linear Systems. New York: Academic Press, 1971.