IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 7.047 ⚟ Vol. 10, Issue 2, February 2022

DOI: 10.17148/IJIREEICE.2022.10212

Illustration of Gauss – Seidel Method Using Matlab

Satakshi Shukla

Student, Electrical Engineering, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune

Abstract: A solution to a linear system in numerical linear algebra is an assignment of values to the variables that concurrently solves all the equations in the system. One of a series of iterative The Gauss–Euler technique is used to solve a linear system of equations. The Liebman technique, sometimes known as the Seidel method, is a method developed by Seidel. Consecutive displacement is a way of displacing objects in a sequential manner. It takes its name from the Carl Friedrich Gauss and Philipp Gauss were German mathematicians. Ludwig von Seidel is a Jacobian scholar who is comparable to Ludwig von Seidel. method. This publication also includes a MATLAB code for solving the problem. The Gauss–Seidel technique is used to solve a linear system of equations numerically.

Keywords: System of linear equations, Gauss-Seidel Method, MATLAB solutions

I. INTRODUCTION

A. MATLAB

MATLAB is a very powerful software package that has many built-in tools for solving problems and for graphical illustrations (1). The name MATLAB stands for Matrix Laboratory (2). The heart of MATLAB is the MATLAB language, a matrix-based language allowing the most natural expression of computational mathematics. The package MATLAB provides an environment in which we can learn to programme and explore the structure of the numerical methods.

Keywords: System of linear equations, Gauss-Seidel Method, MATLAB solutions

B. Gauss–Seidel method

The user can adjust round-off error using the Gauss-Seidel Method. The majority of elimination strategies are prone to round-off mistake. Jacobi's iteration approach is modified in this manner. It is defined on matrices with non-zero diagonals, but only diagonally dominating or symmetric positive definite matrices ensure convergence (3). Furthermore, if the problem's mechanics are known, a near first guess may be made, reducing the number of iterations required.

II. BASIC PROCEDURE:

Solve each linear equation algebraically for the unknowns, such as xi.
Assume a guess solution array for the given problem. linear equations system
Solve for each unknown xi and then repeat the procedure.
After each, use absolute relative approximation error.
iteration to see whether the error is within a pre-defined range.

Algorithm:

Assume we have an initial estimate solution array for the solution array x of the provided system of linear equations Ax = B, such as x (0) = (x1 (0), x2 (0),..., xn (0).

$\sum aij\ xi\ n\ i{=}1 = bj$, for $j = 1, \dots , $ n

From the prior solution estimate x (k), we produce an enhanced solution estimate x (k+1). The Gauss-Seidel approach varies from the Jacobi method in that the available values of xj (k+1) are utilised to update the solution at the (k + 1)-th step, resulting in a better solution estimate x (k+1).

$xj\ (k{+}1) = 1\ ajj\ [bj - \sum aij\ xi\ n\ (k)\ i{=}0\ i{\neq}j\ ]$,

for $j = 1, \dots , n$ (4)

Equation 1. Iteration formula of Gauss–Seidel method

At the end of each cycle, the absolute relative approximation inaccuracy for each xi is calculated as follows:

$|\varepsilon a\ |\ i = |\ xi$ new$-xi$ old/ xi new $|\ x100$

Equation 2. Error for Gauss-Seidel Method

DOI: 10.17148/IJIREEICE.2022.10212

A. MATLAB Algorithm

- Go to the M-files folder and create a new one.
- Input the A and B coefficient matrices.
- Input your initial prediction for the solution of the given system of linear equations, as well as the number of iterations you want to run.
- If the diagonal elements of the matrix A are all zero, display the output and use the Gauss– Seidel method.
- If the matrix A is strictly diagonally dominating and positive definite, the Gauss – Seidel iteration approach ensures that the numerical solution for the linear system converges to the original solution for any initial starting vector.
- Then use the Gauss– Seidel method's Equation 1. Iteration formula to acquire a better estimate of the answer.
- Then use Equation 2. Error for Gauss-Seidel Method to get the absolute relative approximation error at each iteration.
- Save the file and execute it

III. MATLAB PROGRAM CODE

```
%Gauss Seidal
Clc
 format compact
A=input('Enter the Coeficient Matrix A: ');
B=input('Enter the solution Matrix b (column matrix):
 ');
C=[A B]; n=input('Enter the number of iteration:
 ');
X=input('Enter the initial guess (Column matrix):
 ');
x(1)=X(1,1);
y(1)=X(2,1);
z(1)=X(3,1);
for j=1:3 %checking for zero in diagonal entries
diag=0;
if A(j,j)==0
 diag=1;
end
end
 if diag==1
disp('Error! Please check the input, diagonal elements of the matrix cannot be zero')
else
eig_A = eig(A); %Checking for positive definite
positive = 0;
for i = 1:rank(A)
if eig_A(i) <= 0
positive = 1;
end
end
diagdom=0; %Checking for diagonally dominant
for j=1:length(B)
if abs(A(j,j))>= sum(abs(A(j,:)))-abs(A(j,j))
diagdom=1;
end
end
for i=1:n %Iterative step
x(i+1)=(1/C(1,1))*(C(1,4)-C(1,2)*y(i)-
C(1,3)*z(i));
y(i+1)=(1/C(2,2))*(C(2,4)-C(2,1)*x(i+1)-
C(2,3)*z(i));
z(i+1)=(1/C(3,3))*(C(3,4)-C(3,1)*x(i+1)-
C(3,2)*y(i+1));
error1=abs((x(i)-x(i+1))/x(i+1))*100;
error2=abs((y(i)-y(i+1))/y(i+1))*100;
```

```
error3=abs((z(i)-z(i+1))/z(i+1))*100;
end
end
if positive == 1
disp('the matrix is not positive definite hence we can not expect convergence. ')
else if diagdom==1
disp('the matrix is positive definite and diagonally dominant hence, we can expect convergence.')
else
disp('the matrix is positive definite but not diagonally dominant hence we can not expect convergence. ')
 end
 end
error=[error1; error2; error3];
disp('a1= ')
 disp(x(i+1))
disp('a2= ')
disp(y(i+1))
disp('a3= ')
disp(z(i+1))
disp('Absolute relative approximate error for each a in
 %')
 disp(error)
```

CONCLUSION

In this research, we created MATLAB code to solve the general linear system of three equations numerically using Gauss Seidel. Manipulation of the code is simple. perform for any natural number for a linear system of n equation’s. We've also created a function to check for positive definite and negative definite. Inside the above, diagonally dominating

REFERENCES

1. Attaway, Stormy. Matlab: A Practical Introduction to Programming and Problem Solving. Canada : Elsevier, Inc., 2009.
2. Gilat, Amos. MATLAB An Introduction with Applications. Columbia, Ohio : John Wiley & Sons, Inc., 2011.
3. Convergence of Jacobi and Gauss-Seidel Method and Error Reduction Factor. HarpinderKaur, KhushpreetKaur. 2, Department of Mathematics Baba FaridCollege, Bathinda : IOSR Journal of Mathematics (IOSRJM), 2012, Vol. 2.
4. Saleri, Alfio QuarteroniRiccardo SaccoFausto. Numerical Mathematics. New York, NY : Springer, 2011. ISBN 978-1-4757-7394-1.
5. Kaw, Autar K, Kalu, Egwalu E and Nguyen, Duc. Numerical Methods with Applications: Abridged. Florida : University of South Florida, 2011.
6. M.K. Jain, S.R.K. Iyengar, R.K. Jain. Numerical methods for Scientific and Engineering computaion. New Delhi : New Age International (P) Limited, Publishers