Iterative Methods for Solving Systems of Linear Equations

Lecture Notes

Mohammad Sabawi

Department of Mathematics
College of Education for Women
Tikrit University

Email: mohammad.sabawi@tu.edu.iq

01 October 2021

Chapter 1

Solving Systems of Linear Equations

1.1 Iterative Methods

Direct methods are more efficient in solving linear systems of small dimensions in less computational cost than iterative methods. For large linear systems in particular for sparse linear systems iterative methods are more efficient for solving linear systems in terms of computational cost and effort compared to direct methods. In this section we will study the most common and basic iterative methods for solving linear algebraic systems which are **Jacobi method** and **Gauss-Siedel method**.

1.1.1 Jacobi Method

The general form of **Jacobi iterative method** for solving the ith equation in the linear system $AX = B$ for unknown $x_i, i = 1, , \cdots , n$ is:

$$x_i^k = \sum_{j=1}^{n} \Big(-\frac{a_{ij} x_j^{k-1}}{a_{ii}} \Big) + \frac{b_i}{a_{ii}},\ j \neq i,\ a_{ii} \neq 0,\ \text{for } i = 1, \cdots , n,\ k = 1, \cdots , n.$$

It is also known as **Jacobi iterative process** or **Jacobi iterative technique**

Example 1. *Solve the following linear system using Jacobi iterative method*

$$\begin{aligned} 2x_1 + x_2 + x_3 &= 0 \\ x_1 + 3x_2 + x_3 &= 0.5 \\ x_1 + x_2 + 2.5x_3 &= 0 \end{aligned}$$

Solution: These equations can be written in the form

$$x_1 = \frac{-x_2 - x_3}{2},$$
$$x_2 = \frac{0.5 - x_1 - x_3}{3},$$
$$x_3 = \frac{-x_1 - x_2}{2.5}.$$

Writing these equations in iterative form

$$x_1^{k+1} = \frac{-x_2^k - x_3^k}{2},$$
$$x_2^{k+1} = \frac{0.5 - x_1^k - x_3^k}{3},$$
$$x_3^{k+1} = \frac{-x_1^k - x_2^k}{2.5}.$$

Let us start with initial guess $P_0 = (x_1^0, x_2^0, x_3^0) = (0, 0.1, -0.1)$. Substituting these values in the right-hand side of each equation in above to find the new iterations

$$x_1^1 = \frac{-x_2^0 - x_3^0}{2} = \frac{-0.1 - (-0.1)}{2} = \frac{-0.1 + 0.1}{2} = 0,$$
$$x_2^1 = \frac{0.5 - x_1^0 - x_3^0}{3} = \frac{0.5 - 0 - (-0.1)}{3} = 0.2,$$
$$x_3^1 = \frac{-x_1^0 - x_2^0}{2.5} = \frac{-0 - 0.1}{2.5} = -0.04.$$

Now, the new point $P_1 = (x_1^1, x_2^1, x_3^1) = (0, 0.2, -0.04)$ is used in the Jacobi iterative form to find the next approximation P_2

$$x_1^2 = \frac{-x_2^1 - x_3^1}{2} = \frac{-0.2 + 0.04}{2} = \frac{-0.16}{2} = -0.08,$$
$$x_2^2 = \frac{0.5 - x_1^1 - x_3^1}{3} = \frac{0.5 + 0.04}{3} = \frac{0.54}{3} = 0.18,$$
$$x_3^2 = \frac{-x_1^1 - x_2^1}{2.5} = \frac{-0 - 0.2}{2.5} = \frac{-0.2}{2.5} = -0.08.$$

The new point $P_2 = (x_1^2, x_2^2, x_3^2) = (-0.08, 0.18, -0.08)$ is closer to the solution than P_0 and P_1 and is used to find P_3

$$x_1^3 = \frac{-x_2^2 - x_3^2}{2} = \frac{-0.18 + 0.08}{2} = \frac{-0.1}{2} = -0.05,$$
$$x_2^3 = \frac{0.5 - x_1^2 - x_3^2}{3} = \frac{0.5 + 0.08 + 0.08}{3} = \frac{0.66}{3} = 0.22,$$
$$x_3^3 = \frac{-x_1^2 - x_2^2}{2.5} = \frac{0.08 - 0.18}{2.5} = \frac{-0.1}{2.5} = -0.04.$$

This Jacobi iteration process generates a sequence of points $\{P_n\} = \{(x_1^n, x_2^n, x_3^n)\}$ that converges to the solution $(x_1, x_2, x_3) = (-3/38, 4/19, -1/19) = (-0.078947368421053, 0.210526315789474, -0.052631578947368)$. The outline of the results is given in the Table 1.1.

n	x_1^n	x_2^n	x_3^n
0	0.000000000000000	0.100000000000000	−0.100000000000000
1	0.000000000000000	0.200000000000000	−0.040000000000000
2	−0.080000000000000	0.180000000000000	−0.080000000000000
3	−0.050000000000000	0.220000000000000	−0.040000000000000
4	−0.090000000000000	0.196666666666667	−0.068000000000000
5	−0.064333333333333	0.219333333333333	−0.042666666666667
6	−0.088333333333333	0.202333333333333	−0.062000000000000
7	−0.070166666666667	0.216777777777778	−0.045600000000000
8	−0.085588888888889	0.205255555555556	−0.058644444444444
9	−0.073305555555556	0.214744444444444	−0.047866666666667
10	−0.083438888888889	0.207057407407407	−0.056575555555556
11	−0.075240925925926	0.213338148148148	−0.049447407407407
12	−0.081945370370370	0.208229444444444	−0.055238888888889
13	−0.076495277777778	0.212394753086420	−0.050513629629630
14	−0.080940561728395	0.209002969135802	−0.054359790123457
15	−0.077321589506173	0.211766783950617	−0.051224962962963
16	−0.080270910493827	0.209515517489712	−0.053778077777778
17	−0.077868719855967	0.211349662757202	−0.051697842798354
18	−0.079825909979424	0.209855520884774	−0.053392377160494
19	−0.078231571862140	0.211072762379973	−0.052011844362140
20	−0.079530459008916	0.210081138741427	−0.053136476207133

Table 1.1: Jacobi Iterative Solution of Example 1

1.1.2 Gauss-Siedel Method

An improvement of Jacobi method can be made by using the recent values x_i^k, $i, k = 1, \cdots, n$, in the calculations once their values are obtained. This improvement is called **Gauss-Siedel iterative method** and its general form for solving the ith equation in the linear system $AX = B$ for unknown $x_i, i = 1, , \cdots, n$ is:

$$x_i^k = \sum_{j=1}^{i-1} \Big(- \frac{a_{ij} x_j^k}{a_{ii}} \Big) + \sum_{j=i+1}^{n} \Big(- \frac{a_{ij} x_j^{k-1}}{a_{ii}} \Big) + \frac{b_i}{a_{ii}}, \ j \neq i, \ a_{ii} \neq 0,$$

for $i = 1, \cdots, n$, and $k = 1, \cdots, n$.

It is also known as **Gauss-Siedel iterative process** or **Gauss-Siedel iterative technique**

Example 2. *Solve the following linear system using Gauss-Siedel iterative method*

$$\begin{aligned} 2x_1 - 4x_2 + x_3 &= -1 \\ x_1 + x_2 + 6x_3 &= 1 \\ 3x_1 + 3x_2 + 5x_3 &= 4 \end{aligned}$$

.

Solution: Rearrange the system in above such that the coefficient matrix is strictly diagonally dominant

$$\begin{aligned} 3x_1 + 3x_2 + 5x_3 &= 4 \\ 2x_1 - 4x_2 + x_3 &= -1 \\ x_1 + x_2 + 6x_3 &= 1 \end{aligned}$$

.

These equations can be written in the form

$$\begin{aligned} x_1 &= \frac{4 - 3x_2 - 5x_3}{3}, \\ x_2 = \frac{-1 - 2x_1 - x_3}{-4} &= \frac{1 + 2x_1 + x_3}{4}, \\ x_3 &= \frac{1 - x_1 - x_2}{6}. \end{aligned}$$

This suggests the following Gauss-Siedel iterative process

$$\begin{aligned} x_1^{n+1} &= \frac{4 - 3x_2^n - 5x_3^n}{3}, \\ x_2^{n+1} &= \frac{1 + 2x_1^{n+1} + x_3^n}{4}, \\ x_3^{n+1} &= \frac{1 - x_1^{n+1} - x_2^{n+1}}{6}. \end{aligned}$$

We start with initial guess $P_0 = (x_1^0, x_2^0, x_3^0) = (1, 0.1, -1)$. Substitute $x_2^0 = 0.1$ and $x_3^0 = -1$ in the first equation and have

$$x_1^1 = \frac{4 - 3x_2^0 - 5x_3^0}{3} = \frac{4 - 3(0.1) - 5(-1)}{3} = \frac{8.7}{3} = 2.9.$$

Then, substitute the new value $x_1^1 = 2.9$ and $x_3^0 = -1$ into the second equation to obtain

$$x_2^1 = \frac{1 + 2x_1^1 + x_3^0}{4} = \frac{1 + 2(2.9) + (-1)}{4} = 1.45.$$

Finally, substitute the new values $x_1^1 = 2.9$ and $x_2^1 = 1.45$ in the third equation and get

$$x_3^1 = \frac{1 - x_1^1 - x_2^1}{6} = \frac{1 - 2.9 - 1.45}{6} = \frac{-3.35}{6} = -0.558333333333333.$$

Now, we have the now point $P_1 = (x_1^1, x_2^1, x_3^1) = (2.9, 1.45, -0.558333333333333)$ is used to find the next approximation P_2.

Substitute $x_2^1 = 1.45$ and $x_3^1 = -0.558333333333333$ in the first equation and get

$$\begin{aligned} x_1^2 &= \frac{4 - 3x_2^1 - 5x_3^1}{3} = \frac{4 - 3(1.45) - 5(-0.558333333333333)}{3} \\ &= \frac{2.441666666666666}{3} = 0.813888888888889. \end{aligned}$$

Then, substitute the new value $x_2^1 = 0.813888888888889$ and $x_3^1 = -0.558333333333333$ into the second equation to obtain

$$\begin{aligned} x_2^2 &= \frac{1 + 2x_1^2 + x_3^1}{4} = \frac{1 + 2(0.813888888888889) + (-0.558333333333333)}{4} \\ &= \frac{2.069444444444445}{4} = 0.517361111111111. \end{aligned}$$

Finally, substitute the new values $x_2^1 = 0.813888888888889$ and $x_2^2 = 0.517361111111111$ in the third equation and get

$$\begin{aligned} x_3^2 &= \frac{1 - x_1^2 - x_2^2}{6} = \frac{1 - 0.813888888888889 - 0.517361111111111}{6} \\ &= \frac{-0.331250000000000}{6} = -0.055208333333333. \end{aligned}$$

This iteration process generates a sequence of points $\{P_n\} = \{(x_1^n, x_2^n, x_3^n)\}$ that converges to the solution $(x_1, x_2, x_3) = (32/39, 25/39, -1/13) = (0.820512820512820, 0.641025641025641, -0.076923076923077)$. The results are given in the Table 1.2.

n	x_1^n	x_2^n	x_3^n
0	1.000000000000000	0.100000000000000	−1.000000000000000
1	2.900000000000000	1.450000000000000	−0.558333333333333
2	0.813888888888889	0.517361111111111	−0.055208333333333
3	0.907986111111111	0.690190972222222	−0.099696180555556
4	0.809302662037037	0.629727285879630	−0.073171657986111
5	0.825558810763889	0.644486490885417	−0.078340883608218
6	0.819414981794946	0.640122269995419	−0.076589541965061
7	0.820860299946349	0.641282764481909	−0.077023844071376
8	0.820423642303718	0.640955860134015	−0.076896583739622
9	0.820538446098689	0.641045077114439	−0.076930587202188
10	0.820505901555874	0.641020303977390	−0.076921034255544
11	0.820514753115183	0.641027117993706	−0.076923645184815
12	0.820512290647653	0.641025234027623	−0.076922920779213
13	0.820512967271065	0.641025753440729	−0.076923120118632
14	0.820512780090325	0.641025610015504	−0.076923065017638
15	0.820512831680559	0.641025649585870	−0.076923080211072
16	0.820512817432583	0.641025638663523	−0.076923076016018
17	0.820512821363173	0.641025641677582	−0.076923077173459
18	0.820512820278183	0.641025640845727	−0.076923076853985
19	0.820512820577581	0.641025641075294	−0.076923076942146
20	0.820512820494949	0.641025641011938	−0.076923076917814

Table 1.2: Gauss-Siedel Iterative Solution of Example 2

Exercises

Exercise 1. *Repeat Example 1 with Gauss-Siedel iteration. Compute five iterations and compare them with Jacobi iterations in the same example.*

Exercise 2. *Redo Example 2 with Jacobi iteration. Compute five iterations and compare them with Gauss-Siedel iterations in the same example.*