Communication Efficient Linear Equation Solver Based on the Gauss-Seidel Algorithm

Nelson G. Brasil Jr *, Vinay A. Vaishampayan † and Sueli I. R. Costa *
* Institute of Mathematics, Statistics and Computer Science,
University of Campinas, Sao Paulo, Brazil,
Email: nelson.gbrasil@gmail.com, sueli@ime.unicamp.br
† Department of Engineering Science and Physics,
College of Staten Island, City University of New York, Staten Island - NY, United States,
Email: Vinay.Vaishampayan@csi.cuny.edu

***Abstract*—We consider the problem of iteratively solving a system of linear equations using the Gauss-Seidel method. Our objective is to determine the tradeoff between the error and the amount of communication (measured in bits) in a two-party setup, where each party possesses partial information. Specifically our objective is to solve for $x \in \mathbb{R}^2$, the 2×2 system of linear equations $M\boldsymbol{x} = \boldsymbol{b}$ when matrix M is known to both parties, and each party knows only a single component of $\boldsymbol{b} \in \mathbb{R}^2$. It is required that each party know the approximate solution when communication ends.**

We propose a communication-efficient algorithm, prove convergence and determine the speed of convergence of the error with respect to the number of bits exchanged.

I. Introduction [1]

Consider two nodes $\mathbf{A}$ and $\mathbf{B}$, each one with some information about a linear system $M\boldsymbol{x} = \boldsymbol{b}$, where $M = (m_{ij})_{i,\,j=1,2}$ is a 2×2 real matrix, and $\boldsymbol{b} = (b_1,\, b_2)^\top$, $-1 \leq b_i \leq 1$ and $\boldsymbol{x} \in \Re$. Suppose both $\mathbf{A}$ and $\mathbf{B}$ know the matrix M and each one has some information about the $\boldsymbol{b}$, for example $\mathbf{A}$ and $\mathbf{B}$ know b_1 and b_2, respectively. Our goal in this problem is compute an approximation to the exact solution $\boldsymbol{x}_* = (x_1^*, x_2^*)^\top$ in both nodes, i.e., given $\epsilon > 0$, we want to compute $\boldsymbol{x}$ such that $\|\boldsymbol{x} - \boldsymbol{x}_*\| < \epsilon$. Assuming we are able to send some information between the nodes with some cost by round, we want to compute an approximation using the minimum possible number of bits.

II. Numerical Linear Algebra approach

One approach to solve this problem is to use an adaptation of the well known Gauss-Seidel method [1] to solve a linear system.

Let $M = L + U$, where L is the Lower Triangular part of M (including its main diagonal) and $U = M - L$. This iterative method to solve $M\boldsymbol{x} = \boldsymbol{b}$ is

$$\boldsymbol{x}^{(n+1)} = T\boldsymbol{x}^{(n)} + \boldsymbol{c},$$

where $T = -L^{-1}U$ and $\boldsymbol{c} = L^{-1}\boldsymbol{b}$.

[1]This work of NGB and SIRC was partially supported by FAPESP $2013/25997-7$, CNPq $312926/2013-8$, PDSE-CAPES. This work was initiated during a visit by NGB to the City University of New York in 2017.

and can be seen as a fixed-point method when M is a matrix with some properties to be assumed here – M should be a strictly diagonally dominant matrix, i.e. $|m_{ii}| > |m_{ij}|$.

If we could send information without care about cost, we would proceed the Gauss-Seidel method as follows:

(i) $\imath = 0$, $\boldsymbol{x}^{(0)} = (x_1^{(0)},\, x_2^{(0)})$;

(ii) Node $\mathbf{A}$ starts computing

$$x_1^{(\imath+1)} = \frac{b_1 - m_{12}x_2^{(\imath)}}{m_{11}},$$

and sends $x_1^{(\imath+1)}$ to $\mathbf{B}$;

(iii) Node $\mathbf{B}$ computes

$$x_1^{(\imath+1)} = \frac{b_2 - m_{21}x_1^{(\imath+1)}}{m_{22}},$$

and sends $x_2^{(\imath+1)}$ to $\mathbf{A}$;

(iv) We update $\imath \leftarrow \imath + 1$ and return to (ii) while $\|\boldsymbol{x}^{(\imath)} - \boldsymbol{x}_*\| > \epsilon$ or $\imath < \imath_{max}$.

If each bit that we send has some cost, we can adapt the later procedure in order to send a limited number of bits per round.

(i) $\imath = 0$, $\boldsymbol{x}^{(0)} = (x_1^{(0)},\, x_2^{(0)})$, D a nonnegative integer (typically a power of 2 or 10).

(ii) Node $\mathbf{A}$ starts computing

$$y = \frac{b_1 - m_{12}x_2^{(\imath)}}{m_{11}},$$

search for integer p such that, for a fixed D,

$$\left|y - \frac{p}{D}\right| < \frac{1}{2D},$$

set $x_1^{(\imath+1)} = p/D$ and sends to $\mathbf{B}$;

(iii) Node $\mathbf{B}$ computes

$$z = \frac{b_2 - m_{21}x_1^{(\imath+1)}}{m_{22}},$$

search for integer q such that, for a fixed D,

$$\left|z - \frac{q}{D}\right| < \frac{1}{2D},$$

set $x_2^{(\imath+1)} = q/D$ and sends to $\mathbf{A}$;

(iv) We update $\imath \leftarrow \imath + 1$ and return to (ii) while $\|\boldsymbol{x}^{(\imath)} - \boldsymbol{x}_*\| > \epsilon$ or $\imath < \imath_{max}$.

For now and on we will call $\boldsymbol{x}_{GS}$ and $\boldsymbol{x}$ the Gauss-Seidel and our iteration respectively to avoid ambiguity. To establish a relation between $\boldsymbol{x}$ and $\boldsymbol{x}_{GS}$ first we write both iterations in matrix form

$$\boldsymbol{x}_{GS}^{(n+1)} = T\boldsymbol{x}_{GS}^{(n)} + \boldsymbol{c},$$

and

$$\boldsymbol{x}^{(n+1)} = T\boldsymbol{x}^{(n)} + \boldsymbol{c} + S\boldsymbol{\delta}^{(n)},$$

where $\boldsymbol{\delta}^{(n)} = (\Delta_1(n), \Delta_2(n))^\top$ is the error when rationalization step is done and

$$S = \begin{pmatrix} 1 & 0 \\ -m_{21}/m_{22} & 1 \end{pmatrix}.$$

Note that, when $n = 0$,

$$\boldsymbol{x}_{GS}^{(1)} = T\boldsymbol{x}_{GS}^{(0)} + \boldsymbol{c}, \quad \boldsymbol{x}^{(1)} = T\boldsymbol{x}^{(0)} + \boldsymbol{c} + \boldsymbol{\delta}^{(0)} = \boldsymbol{x}_{GS}^{(1)} + \boldsymbol{\delta}^{(0)},$$

since $\boldsymbol{x}^{(0)} = \boldsymbol{x}_{GS}^{(0)}$.

Using the same argument

$$\begin{aligned}\boldsymbol{x}^{(2)} &= T\boldsymbol{x}^{(1)} + \boldsymbol{c} + S\boldsymbol{\delta}^{(1)} \\ &= T(\boldsymbol{x}_{GS}^{(1)} + S\boldsymbol{\delta}^{(0)}) + \boldsymbol{c} + S\boldsymbol{\delta}^{(1)} \\ &= \boldsymbol{x}_{GS}^{(2)} + TS\boldsymbol{\delta}^{(0)} + S\boldsymbol{\delta}^{(1)}.\end{aligned}$$

Applying some inductive process we can show that

$$\boldsymbol{x}^{(n+1)} = \boldsymbol{x}_{GS}^{(n+1)} + \sum_{i=0}^{n} T^i S\boldsymbol{\delta}^{(n-i)}, \text{ for all } n \geq 0. \tag{1}$$

And from the triangular inequality

$$\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\| \leq \|\boldsymbol{x}^{(n)} - \boldsymbol{x}_{GS}^{(n)}\| + \|\boldsymbol{x}_{GS}^{(n)} - \boldsymbol{x}_*\|. \tag{2}$$

Proposition 1. *If M is a strictly diagonally dominant matrix, we can assure that there is an upper bound to the error $\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\|$ when n grows.*

Proof. Starting from the exact solution for the linear system

$$\boldsymbol{x}_* = \left(\frac{m_{22}b_1 - m_{12}b_2}{m_{11}m_{22} - m_{12}m_{21}}, \frac{m_{11}b_2 - m_{21}b_2}{m_{11}m_{22} - m_{12}m_{21}}\right)^\top, \tag{3}$$

we can write the n-th Gauss-Seidel iteration as

$$\begin{aligned}&\boldsymbol{x}_{GS}^{(n)} = T^n\boldsymbol{x}^{(0)} + \sum_{i=0}^{n-1} T^i\boldsymbol{c} = \\ &\left(\frac{m_{22}K^n(m_{21}(b_1 + m_{12}x_2^{(0)}) - m_{11}(b_2 + m_{22}x_2^{(0)}))}{m_{21}(m_{12}m_{21} - m_{11}m_{22})}\right. \\ &\quad + \frac{m_{21}(m_{12}b_2 - m_{22}b_1)}{m_{21}(m_{12}m_{21} - m_{11}m_{22})}, \\ &\frac{K^n(m_{21}(m_{12}x_2^{(0)} - b_1) + m_{11}(b_2 - m_{22}x_2^{(0)}))}{m_{12}m_{21} - m_{11}m_{22}} \\ &\quad \left.\frac{-m_{11}b_2 + m_{21}b_1}{m_{12}m_{21} - m_{11}m_{22}}\right)^\top\end{aligned} \tag{4}$$

where $K = \frac{m_{12}m_{21}}{m_{11}m_{22}}$.

Finally, subtracting (3) of (4), taking the norm and taking (by choice) $x_2^{(0)} = 0$, we have

$$\|\boldsymbol{x}_{GS}^{(n)} - \boldsymbol{x}_*\| = \left|\frac{m_{12}m_{21}}{m_{11}m_{22}}\right|^n \underbrace{\left|\frac{m_{21}b_1 - m_{11}b_2}{m_{11}m_{22} - m_{12}m_{21}}\right|}_{\gamma} \sqrt{1 + \left(\frac{m_{22}}{m_{21}}\right)^2}. \tag{5}$$

To put a bound on the first part of (2), we use (1)

$$\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_{GS}^{(n)}\| \leq \sum_{i=0}^{n-1} \|T^iS\|\|\boldsymbol{\delta}^{(n-1-i)}\| \leq \underbrace{\sqrt{1 + \left(\frac{m_{22}}{m_{21}}\right)^2}}_{\alpha} \sum_{i=0}^{n-1} \underbrace{\left|\frac{m_{12}m_{21}}{m_{11}m_{22}}\right|^i}_{\beta^i} \sqrt{\Delta_1^2(i) + \Delta_2^2(i)}. \tag{6}$$

Combining (5) and (6) in (2) we have

$$\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\| \leq \alpha\left(\beta^n\gamma + \sum_{i=0}^{n-1}\beta^i\sqrt{\Delta_1^2(i) + \Delta_2^2(i)}\right). \tag{7}$$

Assume that we will run the algorithm for n steps and choose the Δ's so that at the end of nth step,

$$\beta^n\gamma = \nu\sum_{i=0}^{n-1}\beta^i\sqrt{\Delta_1^2(i) + \Delta_2^2(i)} \tag{8}$$

for some previously chosen constant ν and (7) becomes

$$\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\| \leq \alpha(1 + \nu)\beta^n\gamma.$$

$\square$

Now, let R_n be the total number of bits exchanged. Suppose $\Delta_1(i) = \Delta_2(i) = \Delta(i)$. Then at the ith step we exchange $R(i) = 2\log_2(1/\Delta(i))$ bits and $R_n = \sum_{i=1}^{n} R(i)$ is the total number of bits.

Proposition 2. *The best choice of $\Delta(i)$ in order to minimize R_n satisfying (8) for some constant ν, is*

$$\Delta(i) = \frac{\beta^{(n-i)}\gamma}{\sqrt{2}n\nu}. \tag{9}$$

Note that minimize the total number of bits is the same of maximize the products of $\Delta(i)$ for all i. Using the Lagrange multipliers we obtain (9).

Example 3. *Consider the following four matrices:*

$$M_1 = \begin{pmatrix} 10 & 1 \\ 2 & 8 \end{pmatrix}, M_2 = \begin{pmatrix} 2 & 1 \\ -1 & 3 \end{pmatrix},$$

$$M_3 = \begin{pmatrix} 1 & 0.9 \\ 0.9 & 1 \end{pmatrix}, M_4 = \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix}.$$

Matrices M_1 and M_2 are strongly diagonal dominant, M_3 is almost not diagonally dominant and M_4 the diagonal elements are smaller than the others.

For each matrix we run the proposed Modified Gauss Seidel Method using the choice of $\Delta(i) = 1/D_i$ based on Proposition 2. Also, we choose three values for $\nu = 10^{-5}$ (blue dots), 10^{-6} (yellow dots) and 10^{-7} (green dots) based on previous tests. The behavior can be seen in Figure 1 to Figure 4. It is important to note that we choose the same solution for all linear systems.

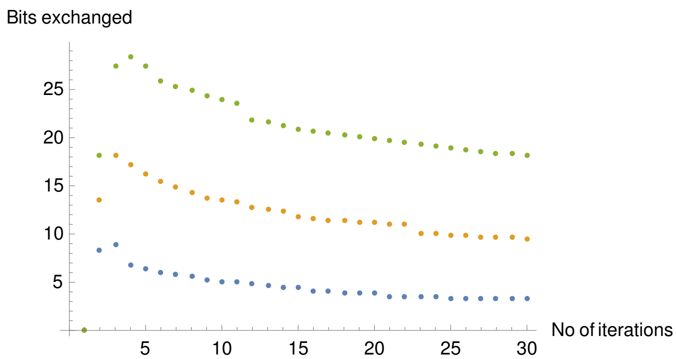

Fig. 1. Matrix M_1. Maximum error$\approx 10^{-2}$

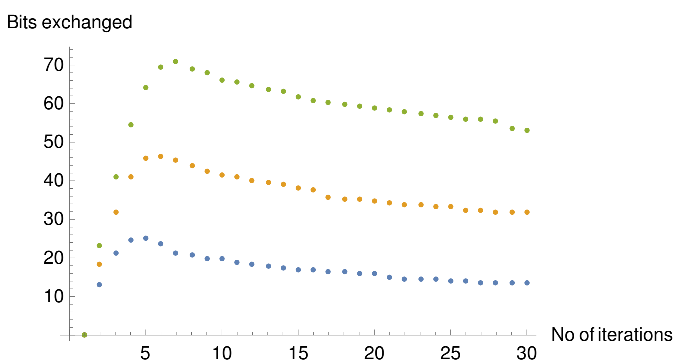

Fig. 2. Matrix M_2. Maximum error$\approx 10^{-3}$

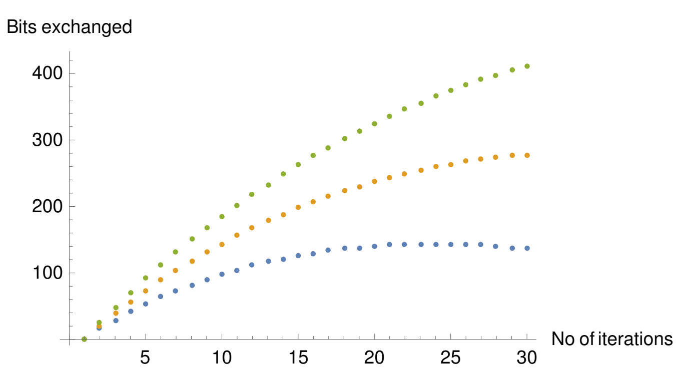

Fig. 3. Matrix M_3. Maximum error$\approx 10^{-4}$

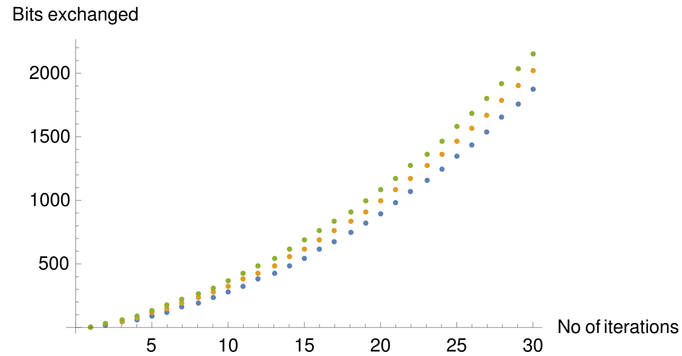

Fig. 4. Matrix M_4. The method does not converge.

III. Conclusion

In this work, we proposed a adaptation of the Gauss-Seidel method to solve a linear system involving two nodes exchanging some information. We developed a method that converges to the original solution under some assumptions.

A natural extension to be approached in a future work is to consider a bigger number of nodes exchanging information resulting in a linear system of higher order and analyze the method we proposed here. In dimension bigger than two, it is possible to analyze two different situation who leads in two known iterative methods: The Gauss-Seidel Method or the Jacobi method.

References

[1] C. D. Meyer, *Matrix analysis and applied linear algebra,* Siam, 2000.