GLOBAL JOURNAL OF MATHEMATICAL SCIENCES VOL10, NO.1 & 2, 2011: 9-15

 ISSN 1596-6208
www.globaljournalseries.com; Email: info@globaljournalseries.com

BOUNDING THE ERROR OF A CONTINUOUS APPROXIMATION FOR LINEAR SYSTEMS

STEPHEN EHIDIAMHEN UWAMUSI

(Received 12, October 2010; Revision Accepted 8, March 2011)

ABSTRACT

We present preconditioned interval Gauss-Siedel method and interval LU decomposition for finding solution to the interval linear system of equation Ad=b where the nxn coefficient matrix A lies between two bounds $\underline{A}$ *and* $\overline{A}$ *and* $b \in [\underline{b}, \overline{b}]$. It is found out that preconditioned interval methods of Gauss-Siedel and LU have substantial reduction of excess widths of the interval hull of the solution set. In particular we also give our results in terms of midpoint-radius arithmetic for Gauss-Siedel method in the sense analogous to (Rump,1999) and (Gargantini and Henrici,1972) circular interval arithmetic.

KEY WORDS: linear systems, LU factorization Gauss-Siedel method, preconditioning matrix, reliable computing

INTRODUCTION

Across all branches of Engineering and Sciences, computational methods provide the quest for reliable results. Reliability is achieved only if all sources of errors, approximations and uncertainty are accounted for (Hayes, 2003), (Moore, 1966) and (Rump, 1999) are good references behind this theory. Measurements are always not 100% accurate. Any one working in Engineering discipline, Physical Sciences, Technical discipline will surely inquire about the effect of rounding error and propagated error due to inexact initial data or uncertain values of parameters in any mathematical models, (Kreinovich and Longpre,2004). To describe this, consider the measurement $\tilde{x}_i$ made by a manufacturer of an equipment. Due to this measurement error defined as $\Delta x_i = \tilde{x}_i - x_i$, the image $\tilde{y} = F(\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n)$ of data processing became generally different from the actual error $\Delta = \tilde{y} - y$ of the result of the data processing y. With this we are able to understand some information about errors of direct measurement. Assuming we perform a measurement and obtain a measurement result $\tilde{x}_i$. We can find the exact (unknown) value $\tilde{x}_i$ of the measured quantity which belongs to the interval $\left[\underline{x}_i, \overline{x}_i\right]$, where $\underline{x}_i = \tilde{x} - \Delta_i$, $\overline{x}_i = \tilde{x}_i + \Delta_i$.

Let $[\underline{x}]$ be replaced by $[a] = [a_1, b_1]$ *and* $\bar{x}$ be replaced by $[b] = [a_2, b_2] \in IR$ *where* IR is the set of intervals with real components, then the properties of interval arithmetic operations can be found in (Alefeld and Herzberger,1983), (Kreinovich and Longpre,2004), (Moore,1966), (Neumaier,1990 and 1986) as follows:

$$a + b = [a_1 + a_2, b_1 + b_2],$$

$$a - b = [a_1 - b_2, a_2 - b_1],$$

$$a \times b = [\min(a_1 a_2, ..., b_1 b_2), \max(a_1 a_2, ..., b_1 b_2)]$$

$$\frac{a}{b} = [a_1, b_1] \times \left[\frac{1}{a_2}, \frac{1}{b_2}\right], b \neq 0$$

The use of interval arithmetic has some important advantages in numerical computing. See for example, (Gau and Stadtherr,2002), (Oishi and Rump,2002), (Kearfott,1996) and the cited references therein. However, the operations of this interval arithmetic are so delicate that wrong interpretations can lead to utterly wrong results. Other reason is that interval arithmetic cannot eliminate round off errors, but it can fence it in. Thus when a result d falls between two floating point values, those nearest representable numbers become the lower and upper bounds of the interval $[\underline{d}, \overline{d}]$.

Stephen Ehidiamhen Uwamusi, Department of Mathematics, University of Benin, Edo State, Nigeria

However, subsequent computations could yield a new interval for which $\underline{d}$ and $\overline{d}$ are themselves numbers that have no floating point representation (cf, Rump, 2001) and (Hayes, 2003).

Another area where interval arithmetic distinguishes itself from ordinary floating point arithmetic is that in general, an interval has no additive inverse, that is, given a non degenerate interval $\left[\underline{d}, \overline{d}\right]$ there is no interval for which $\left[\underline{d}, \overline{d}\right] + \left[\underline{c}, \overline{c}\right] = [0,0]$. Interval arithmetic does not possess multiplicative inverse too, that is there exists no pair of degenerate intervals for which $\left[\underline{d}, \overline{d}\right] \times \left[\underline{c}, \overline{c}\right] = [1,1]$.

Since we will be dealing mostly with interval vectors and interval matrices, then we define $R^n, R^{nxn}, IR, IR^n, IR^{nxn}$ to signify the set of real vectors with n components, the set of real nxn matrices, the set of intervals, the set of interval vectors with components and the set of nxn interval matrices, respectively (Kearfott 1996) and, (Neumaier 1990).

Consider a given interval linear system in the form:
Ad=b (1.1)
where

$$A\tilde{d} = [\underline{A}\tilde{d}, \overline{A}\tilde{d}] \qquad (1.2)$$

In (1.2), we have expressed $A\tilde{d}$ in terms of end points of elements of A because we know the signs of the components of d. Thus we have a system of interval linear equation in the form

$$A\tilde{d} = \left[\underline{A}\tilde{d}, \overline{A}\tilde{d}\right] = \left[\underline{b}, \overline{b}\right] \qquad (1.3)$$

We expect that the variable $\tilde{d}$ must be such that the intervals intersect. It follows that

$$\underline{A}\tilde{d} \geq \underline{b} \text{ and } \overline{A}\tilde{d} \leq \overline{b} \qquad (1.4)$$

One assertion of interval arithmetic is that it can be used to test naturally the Brouwer fixed point theorem, (Ning and Kearfott,1997). The Brouwer fixed point theorem in interval arithmetic asserts that , if ID is a homeomorphism to the closed unit ball in IR^n and G is a continuous mapping such that G maps ID into ID, then there is $d \in IR^n$ for which d=G(d).

Rump's interval matrix operations

Rump's operations as defined in (Rump,1999) on interval matrix are quite similar to circular interval arithmetic introduced by (Gargantini and Henrici,1972). In these formulas every interval $[a] = [a_1, a_2]$ is represented by its midpoint $a_c = \left(\frac{a_1 + a_2}{2}\right)$ and its half-width (radius) $r = \left(\frac{a_1 - a_2}{2}\right)$, thus $a = [a_c - r, a_c + r]$. The corresponding arithmetic operations for two intervals $a = [a_c - r_1, a_c + r_1], b = [b_c - r_2, b_c + r_2]$ will now take the form given by

$$[a_c - r_1, a_c + r_1] \circ [b_c - r_2, b_c + r_2] = [c_c - r_3, c_c + r_3]$$

where

$$a \circ b = a + b, we\ have\ c_c = a_c + b_c\ and\ r_3 = r_1 + r_2,$$
$$a \circ b = a - b, we\ have\ c_c = a_c - b_c\ and\ r_3 = r_1 + r_2,$$
$$a \circ b = a \cdot b, we\ have\ c_c = a_c \cdot b_c\ and\ r_3 = |a_c| \cdot r_2 + |b_c| \cdot r_1 + r_1 \cdot r_2$$

We also define the inverse disk $(a_c, r)^{-1}$ to be $\left(\frac{a_c}{a_c^2 - r^2}, \frac{r}{a_c^2 - r^2}\right)$.

The purpose of this paper is to show that preconditioning the interval linear system (1.1) before employing the interval Gauss-Siedel method and LU factorization has substantial gains in reduction of excess widths than using crude interval Gauss-Siedel method and LU factorization especially for intervals whose widths are small since experience also showed that it can produce utterly overestimated results when the interval widths are large.

The Methods

In this section we will describe interval Gauss-Siedel method and the LU factorization as approximate solution set to the interval linear system (1.1). For detailed description of Gauss-Siedel method one can consult (Ortega and Rheinboldt, 2000) see also (Ning and Kearfott, 1997), (Alefeld and Herzberger,1983).

In the case of LU factorization one solves a kind of linear system

LUd=b (3.1)

by an explicit splitting as follows:

Forward solve in the following steps:

$$\begin{bmatrix} 1 & \dots & & 0 \\ m_{2,1} & 1 \dots & & 0 \\ \dots & & \dots & \\ m_{n,1} & m_{n,2} \dots & m_{n,n-1} & 1 \end{bmatrix} \begin{pmatrix} z_1 \\ z_2 \\ . \\ z_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ . \\ b_n \end{pmatrix} \qquad (3.2)$$

Backward solve

$$\begin{bmatrix} u_{1,1} & u_{1,2} & \dots & u_{1,n} \\ 0 & u_{2,2} & \dots & u_{2,n} \\ . & . \dots & & .. \\ . & & & \\ . & & & \\ 0 & \dots & \dots & u_{n,n} \end{bmatrix} \begin{pmatrix} d_1 \\ d_2 \\ . \\ . \\ . \\ d_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ . \\ . \\ . \\ y_n \end{pmatrix} \qquad (3.3)$$

where

$m_{i,j}$ are the elementary matrices called the multipliers, the $u_{i,j}(i = 2,..nj = 1,2,....n)$ are the upper triangular elements of the decomposed matrix A.

In compact form the algorithmic structure of LU factorization is given by:

For i=2,…n

Set $z_i = b_i - \sum_{j=1}^{i-1} m_{ij} z_j$

End for loop

For i=n,n-1,…,1

Set $d^{(i)} = \dfrac{z_i - \sum_{t=I+1}^{n} u_{ii} d_t}{u_{ii}}$

End for loop

The error analysis in LU factorization can be seen as follows:

Assuming that d* is an approximate solution to the system of equations (1.1). We consider the problem of calculating the bounds of $\left\|A^{-1}b - d^*\right\|_\infty$ where $\|d\|_\infty$ is the infinity norm in IR^n.

We suppose that there is an approximate inverse matrix B to the interval matrix A together with approximate solution d^* in system (1.1). Multiplying through equation 3.1 by the approximate inverse matrix B will give what is called preconditioned interval LU decomposition. It is known (Ortega and Rheinboldt,2000) that $\|BA - I\|_\infty < 1$ implies that $\rho(BA - I) < 1$ which proves the existence of A^{-1}. Together with application of Perron-Ferobenius theorem, we have that $|I - BA|d < d$

Setting $A = \left(I - (I - BA)^{-1}\right)B$, it can be shown (see e.g., Oishi and Rump,2002) that $\|BA - I\|_\infty \le \dfrac{\|B\|_\infty}{1 - \|BA - I\|_\infty}$

and that $\left\|A^{-1}b - d^*\right\|_\infty \le \dfrac{\|B(Ad^* - b\|_\infty}{1 - \|BA - I\|_\infty}$. (3.4)

We will further assume that LU factorization is given as described above say, with a permutation matrix P in the form $LU \approx PA$. We can then compute the approximate inverses A_L *and* A_U of L and U by replacing B by $A_L A_U P$. Substituting this into equation (3.1), we have the error bound in the form $\left\|A^{-1}b - d^*\right\| \le \dfrac{\theta}{1 - \beta}$ where we set $\theta = \|A_U A_L PA - I\|_\infty$, $\beta = \|A_U A_L P(AD^* - b)\|_\infty$

Employing interval Gaussian elimination to system 1.1, we see that interval widths tend to grow. Let us note that the interval Lu factorization described earlier is always obtained from interval Gaussian elimination. One can overcome this defect of wider interval widths if we precondition the interval linear system whereby we multiply the linear interval system Ad=b by an appropriate inverse of the centre of A. The wider the vector b the wider the solution set will be. The closer the inverse midpoint matrix is to an identity matrix, the less the preconditioning step tends to enlarge the solution step. As attempt in solving for the solution to system 1.1, let us note that concept of fixed point mapping $G : ID \subset IR^n \to IR^n$ to be satisfied is essential. Thus for a non singular preconditioning matrix B it is that f(d)=0 if and only if G(d)=d, where G(d)=d-Bf(d) in the sense of Brouwer's Fixed point theorem. On the basis of this, the equation G(d)=Ad=b will be rewritten in the form:

G(d)=d-B(Ad-b) (3.5)

This implies that Bb+(I-BA)d=G(d).The essence of equation 3.5 helps to prove the sufficient conditions for the existence of a fixed point and error analysis in our solution to system 1.1. Let us note that every contraction in $ID \subset IR^n$ is Lipschitz continuous.
As a remark we define the interval Gauss-Siedel iteration in the form

$d^0 = d, d^{l+1} = \Gamma(A, b, d^l)$, (l=0,1,2,..) (3.6)

Thus the preconditioned interval Gauss-Siedel iteration with an approximate real point inverse matrix B of interval matrix A is the equation

$d^0 = d, d^{l+1} = \Gamma(BA, Bb, d^l),\ (l = 0,1,2,..)$ (3.7).

Let us note that for all $l \ge 0$ the components of the Gauss-Siedel iteration (3.6) satisfy

$$d_i^{l+1} = \Gamma\left(A_{ii}, b_i - \sum_{k<i} A_{ik} d_k^{l+1} - \sum_{k>i} A_{ik} d_k^{l}, d_i \right), i = 1,2,...,n \quad (3.8)$$

Where Γ is a graph connecting i and k.

Numerical Experiment

As an illustration we will consider the following problem:
Ad=b
Where

A=

$$\begin{pmatrix} [20,20] & [-.0610307587,-0.578140768] & [0.189883127,0.201771786] & [0,0] \\ [1.527685488,1.578081928] & [20,20] & [0,0] & [-0.193054490,-0.153964193] \\ [-0.145876679,-0.106198125] & [-1.145876679,-1.106198125] & [18.50590343,18.54168695] & [0,0] \\ [0,0] & [1.799117812,1.821412458] & [-1.884627872,1.848254165] & [21,21] \end{pmatrix}$$

$$d = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix}$$

$$b = \begin{bmatrix} [2.224514638,2.672484857] \\ [-2.200440106,-1.772842959] \\ [1.7647269,2.173919257] \\ [-1.965396044,-1.471852763] \end{bmatrix}$$

The following table1 gives the result from the application of Interval Gauss-Siedel method without preconditioning.

Table 1

ITERATION	Results from Interval Gauss-Siedel Method (3.8)
1	-0.133624242,-0.111225731 0.097138044,0.1250565505 -0.112500982,-0.088533135 0.049534798,0.077476263
2.	-0.129975713,-0.106411653 0.097248470, 0.1210254780 -0.112465689,-0.088477028 0.049498070, 0.077471740
3.	-0.129973054,-0.106397973 0.097160902,0.121025225 -0.112470892,-0.088736456 0.049497625, 0.077455564
4.	-0.129973122,-0.106397928 0.097150319,0.121025074 -0.112471524, -0.088476974 0.049497581, 0.077480154

We halt iteration after four successive complete cycles. The result for LU factorization method is given in table 2.

Table 2: Interval Lu Factorization (3.1)

d_i	RESULTS
d_1	-0.129907622, - 0.106346463
d_2	0.097371623, 0.120952314
d_3	-0.117795459, -0.09499746
d_4	0.03963671, 0.077786746

Table 3: (Results for Preconditioned Interval Lu Factorization (3.1))

d_i	RESULTS
d_1	-0.129710833, - 0.106693078
d_2	0.098909935, 0.119201072
d_3	-0.112379935, -0.088409123
d_4	0.029609098, 0.059380416

Table 4: (Preconditioned Interval Gauss-Siedel Method (3.7))

ITERATION	RESULTS
1	-0.129581663, -0.106839440 0.098974170, 0.119136727 -0.112335277, -0.088531191 0.029655320,0.059745253
2.	-0.129628995, - 0.106695347 0.098909425, 0.119205537 -0.112337144, -0.088527517 0.029390932, 0.059361003
3.	-0.129626397, -0.106768289 0.098858662, 0.118743692 -0.112376431, - 0.088528117 0.029426214, 0.059245068

Table 5: (Applied Rump's operation on Gauss-Siedel method (3.8))

ITERATION	RESULTS IN MID POINT-RADIUS INTERVALS.
1	-0.122424987, -0.011199255 0.108837665, 0.011419351 -0.100530134, -0.011825395 0.063522181, 0.011677008
2	-0.112578053, - 0.010335898 0.108624192, 0.011391449 -0.100476112, - 0.010104468 0.063545384, 0.011976915
3	-0.118213827, -0.010825516 0.109061978, 0.0098645226 -0.100487846, -0.010189541 0.063506604, 0.012105594
4	-0.118200705, 0.000315788 0.10906062, 0.011042682 -0.100487839, -0.010028782 0.06350672, 0.011953699

CONCLUSION

From results presented in Tables 1-4, it can be observed that those of preconditioned interval Gauss-Siedel (3.7) and preconditioned LU factorization method (3.1) have substantial reduction in excess widths in the solution hull to system 1.1 which are shown in Tables 3 and 4 wherein we implemented (Moore,1966) version of interval arithmetic in Tables 1-4. Practically, is the simultaneous construction of two sided converging sequences to their respective limits taking advantage of outward rounding wherein one is the sequence of lower bounds on the enclosures converging to the range infimum, and the other is the sequence of upper bounds on the enclosures converging to the range supremum. This is in sharp contrast to the midpoint-radius interval results presented in table 5 for Gauss-Siedel method without preconditioning.

REFERENCES

Alefeld, G., and Herzberger, J. 1983. Introduction to Interval Computation. Academic Press, New York.

Gau,C.Y., and Stadtherr,M.A., 2002. New Interval Methodologies for reliable chemical Y. Process. Modelling, Comput. Chem. Eng., (26): 827-840.

Gargantini,I., and Henrici, P., 1972. Circular Arithmetic and Determination of Polynomial Zeros. Numer. Math., (18): 305-320.

Hayes, B., 2003. A Lucid Interval. American Scientist, (91): 484-488.

Kearfott, R. B., 1996. Interval Computations: Introduction, Uses, and Resources. Euromath Bulletin 2, (1): 95-112.

Kearfott, R. B., 1996. Rigorous global search: Continuous problems, Kluwer, Dordrecht, Netherlands.

Krienovich, V.,and Longpre, L., 2004. Fast quantum algorithms for handling probabilistic and interval uncertainty. Mathematical Quarterly, 50, (4/5): 507-518.

Moore, R. E., 1966. Interval Analysis. Englewood Cliffs, NJ: Prentice Hall

Neumaier, A., 1990. Interval Methods for Systems of Equations, Cambridge University Press, Cambridge, England.

Neumaier, A., 1986. On the comparison of H-matrices with M- matrices, Linear Algebral Appl, (83):135-141.

Ning, S., and Kearfott,R.B., 1997. A comparison of some methods for solving Linear interval equations, SIAM J. Numerical Analysis 34, (1): 1289-1305.

Oishi, S., and Rump, S. M., 2002. Fast verification of solutions of matrix equations. Numer. Math., 90, (4): 755-773.

Ortega, J. M., and Rheinboldt, W. C., 2000. Iterative solution of nonlinear equations in several variables. Classics in Applied Mathematics, SIAM 30.

Rump, S. M., 1999. Fast and Parallel interval arithmetic. BIT 39, (3): 539-560.

Rump, S. M., 2001. Self validating methods. LAA 324, Issue 1(3): 1-13.