2008 International Symposium on Nonlinear Theory and its Applications
NOLTA'08, Budapest, Hungary, September 7-10, 2008

Iterative Refinement for Ill-conditioned Linear Equations.

Shin'ichi Oishi[1,4], Takeshi Ogita[2,4] and Siegfried M. Rump[3,1]

[1]Department of Applied Mathematics, Faculty of Science and Engineering,
Waseda University,
3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555 Japan
[2] Department of Mathematics, College of Arts and Sciences,
Tokyo Woman's Christian University,
2-6-1 Zempukuji, Suginami-ku, Tokyo 167-8585, Japan
[3] Institute for Reliable Computing,
Hamburg University of Technology
Schwarzenbergstr. 95 21071 Hamburg Germany
[4] CREST, JST, Japan

Abstract—This paper treats a linear equation

$$Av = b,$$

where $A \in \mathbb{F}^{n\times n}$ and $b \in \mathbb{F}^n$. Here, $\mathbb{F}$ is a set of floating point numbers. Let $\mathbf{u}$ be the unit round-off of the working precision and $\kappa(A) = \|A\|_\infty \|A^{-1}\|_\infty$ be the condition number of the problem. In this paper, ill-conditioned problems with

$$1 < \mathbf{u}\kappa(A) < \infty$$

are considered and an iterative refinement algorithm for the problems is proposed. In this paper, the forward and backward stability will be shown for this iterative refinement algorithm.

1. Introduction

In this paper, we will consider the convergence of an iterative refinement for a linear equation

$$Av = b, \tag{1}$$

where $A \in \mathbb{F}^{n\times n}$ and $b \in \mathbb{F}^n$. Here, $\mathbb{F}$ is a set of floating point numbers. Let $\mathbf{u}$ be the unit round-off of the working precision and $\kappa(A) = \|A\|_\infty \|A^{-1}\|_\infty$ be the condition number of the problem. Here, $\|\cdot\|_\infty$ is the maximum norm defined by

$$\|v\|_\infty = \max_{1\leqq i\leqq n} |v_i|, \quad \text{for } v = (v_1, v_2, \cdots, v_n)^T \in \mathbb{R}^n \tag{2}$$

and

$$\|A\|_\infty = \max_{1\leqq i\leqq n} \sum_{j=1}^{n} |A_{ij}| \quad \text{for } A = (A_{ij}) \in \mathbb{R}^{n\times n}. \tag{3}$$

The superscript T denotes the transpose and $\mathbb{R}$ is the set of real numbers. For well posed problems, *i.e.*, in case of $\mathbf{u}\kappa(A) < 1$, it has been shown [1]-[5] that the iterative refinement improves the forward and backward errors of computed solutions provided that the residuals are evaluated by extended precision, in which the unit round off $\overline{\mathbf{u}}$ is, for example, the order of $\mathbf{u}^2$, before rounding back to the working precision. In this paper, we will treat ill-conditioned problems with

$$1 < \mathbf{u}\kappa(A) < \infty. \tag{4}$$

We can assume without loss of generality that for a certain positive integer k the following is satisfied:

$$\mathbf{u}^k\kappa(A) \leqq \beta < 1. \tag{5}$$

In Ref. [8], Rump has shown that for arbitrary ill-conditioned matrices A, we can have good approximate inverses $R_{1:k}$ satisfying $\|R_{1:k}A - I\|_\infty \leqq \alpha < 1$. Here, $R_{1:k}$ is obtained as

$$R_{1:k} = R_1 + R_2 + \cdots + R_k \tag{6}$$

with $R_i \in \mathbb{F}^{n\times n}$ and I is the n-dimensional unit matrix. In Ref. [6], we have partially clarified the mechanism of the convergence of Rump's method.

Let $A, B, C \in \mathbb{F}^{n\times n}$. We assume that we have an accurate matrix product calculation algorithm $[AB - C]_k$ such that

$$D_{1:k} = [AB - C]_k \tag{7}$$

satisfying

$$\left\| \sum_{i=1}^{k} D_i - (AB - C) \right\|_\infty \leqq c\mathbf{u}^k \|AB - C\|_\infty. \tag{8}$$

Here, $D_{1:k}$ is defined as

$$D_{1:k} = D_1 + D_2 + \cdots + D_k \tag{9}$$

with $D_i \in \mathbb{F}^{n\times n}$. Such algorithms have been proposed by the present authors with $c < 2.1$ (See [7], [9] and [10]).

Now we propose the following iterative refinement algorithm:

$$v' = [v - R_{1:k}[Av - b]_k]_1. \tag{10}$$

Put $r_k = [Av - b]_k$ and let $\Phi(v) = [v - R_{1:k} r_k]_1$. Then, we can write

$$v' = \Phi(v). \tag{11}$$

The following holds:

$$v' = v - R_{1:k}[(Av - b) + e_r] + e_m, \tag{12}$$

where $e_r = r_k - (Av - b)$ and $e_m \in \mathbb{R}^n$ satisfying

$$\|e_r\|_\infty \leqq c\mathbf{u}^k \|Av - b\|_\infty \tag{13}$$

and

$$\|e_m\|_\infty \leqq c\mathbf{u}\|v - R_{1:k} r_k\|_\infty. \tag{14}$$

In this paper, we will show the forward and backward stability of the iterative algorithm (10). Furthermore, numerical examples are also given for illustrating the forward and backward stability of the iterative refinement algorithm (10). The forward stability of the algorithm guarantees that approximate solutions generated by the algorithm converge, while the backward stability means the stability of the algorithm against the rounding errors.

2. Convergence Theorem: Forward Stability

Let us consider

$$Av = b, \tag{15}$$

where $A \in \mathbb{F}^{n\times n}$ and $b \in \mathbb{F}^n$. Let

$$1 < \mathbf{u}\kappa(A) < \infty. \tag{16}$$

We assume that we have a good approximate inverses $R_{1:k}$ satisfying

$$\|R_{1:k}A - I\|_\infty \leqq \alpha < 1. \tag{17}$$

Here, $R_{1:k}$ is defined as

$$R_{1:k} = R_1 + R_2 + \cdots + R_k \tag{18}$$

with $R_i \in \mathbb{F}^{n\times n}$. As mentioned in the previous section in Ref.[8], Rump has proposed a method of calculating such approximate inverses and in Ref.[6], we have partially clarified the mechanism of the convergence of Rump's method. Further, we assume also that the following is satisfied:

$$\mathbf{u}^k \kappa(A) \leqq \beta < 1. \tag{19}$$

We propose the following iterative refinement algorithm:

$$\begin{aligned} v_n &= \Phi(v_{n-1}), \quad \Phi(v) = [v - R_{1:k} r_k]_1, \\ r_k &= [Av - b]_k \quad (n = 1, 2, \cdots) \end{aligned} \tag{20}$$

with any starting vector $v_0 \in \mathbb{F}^n$. The aim of this section is to show the following theorem:

Theorem 1 *Let v_n be generated from (20) with any starting vector $v_0 \in \mathbb{F}^n$. We assume the assumptions (17) and (19). If*

$$\gamma = (\alpha + c\beta + c\alpha\beta)(1 + c\mathbf{u}) < 1, \tag{21}$$

the relative forward error $\|v_n - v^*\|_\infty / \|v^*\|_\infty$ reduces until

$$\frac{\|v_n - v^*\|_\infty}{\|v^*\|_\infty} \approx \mathbf{u} + \frac{c\mathbf{u}}{1-\gamma}. \tag{22}$$

Here, for real numbers a and b, $a \approx b$ means that a is approximately equal to b.

This implies the forward stability of the iterative refinement algorithm (20).

3. Backward Stability

In this section, we will show the backward stability of the iterative refinement algorithm (20).

A normwise backward error of an approximation v is defined by

$$\begin{aligned} \eta(v) \ = \ & \min\{\varepsilon : (A + \Delta A)v = b + \Delta b, \\ & \|\Delta A\|_\infty \leqq \varepsilon \|A\|_\infty, \\ & \|\Delta b\|_\infty \leqq \varepsilon \|b\|_\infty\}. \end{aligned} \tag{23}$$

It is known [13] that

$$\eta(v) = \frac{\|r\|_\infty}{\|A\|_\infty \|v\|_\infty + \|b\|_\infty}. \tag{24}$$

Here, $r = Av - b$.

The next theorem shows the backward stability of the iterative refinement algorithm (20):

Theorem 2 *Let v_n be generated from (20) with any starting vector $v_0 \in \mathbb{F}^n$. We assume the assumptions (17) and (19). If*

$$\gamma = (\alpha + c\beta + c\alpha\beta)(1 + c\mathbf{u}) < 1, \tag{25}$$

the backward error $\eta(v)$ reduces until

$$\eta(v) \lessapprox c_2\mathbf{u}, \tag{26}$$

where c_2 is a certain constant. Here, for real numbers a and b, $a \lessapprox b$ means that a is approximately equal to b or a is less than b.

This implies the backward stability of the iterative refinement algorithm (20).

4. Numerical Examples Illustrating Forward and Backward Stability

In this section, we will present numerical examples illustrating the forward and the backward stability of the iterative refinement algorithm (20).

We have used the IEEE 754 double precision floating point number system in these numerical calculations. Thus, in the following calculations, the unit round-off $\mathbf{u}$ is given as

$$1.11 \times 10^{-16} < \mathbf{u} = 2^{-53} < 1.12 \times 10^{-16}. \tag{27}$$

4.1. Hilbert Matrix

Let H be the $n \times n$ Hilbert matrix. Let further $A = sH$. Here, s is the minimum common multiplier of $1, 2, 3, \cdots, n-1$. Furthermore,

$$b = Az, \tag{28}$$

where, $z \in \mathbb{F}^n$ and $z_i = 1$. We have solved $Ax = b$ for $n = 20$. In this example, $1.92 \times 10^{16} < \|A\|_\infty < 1.93 \times 10^{16}$, $1.92 \times 10^{16} < \|b\|_\infty < 1.93 \times 10^{16}$ and $2.44 \times 10^{28} < \kappa(A) < 2.45 \times 10^{28}$.

In this case, a good approximate inverse can be constructed with $k = 2$ such that

$$\|RA - I\|_\infty < \alpha = 4.16 \times 10^{-4}, \tag{29}$$

where

$$R = R_1 + R_2 \tag{30}$$

with suitable $R_1, R_2 \in \mathbb{F}$. The iterative refinement algorithm (20) converges with 3 iterations. We finally have an approximate solution with the relative maximum error about 1.92×10^{-16}. Furthermore, it is seen that

$$\begin{aligned} \beta &= \mathbf{u}^2\kappa(A) < (1.2 \times 10^{-16})^2 \times 2.45 \times 10^{28} \\ &< 3.08 \times 10^{-4}. \end{aligned} \tag{31}$$

Table 1 shows the relative errors

$$\frac{\|v^* - v_i\|_\infty}{\|v^*\|_\infty} \tag{32}$$

and the backward errors $\eta(v_i)$ of approximate solutions obtained by the iterative refinement calculations (20). These calculations are done by MATLAB on Intel core 2 duo CPU.

Table 1: Hilbert matrix (n=20)

i	$\|v^* - v_i\|_\infty / \|v^*\|_\infty$	$\eta(v_i)$
0	3.50×10^{-4}	4.55×10^{-6}
1	4.03×10^{-9}	4.12×10^{-11}
2	5.10×10^{-14}	5.04×10^{-16}
3	1.91×10^{-16}	1.77×10^{-18}
4	1.91×10^{-16}	1.77×10^{-18}

4.2. Rump's Matrix (n=100)

Let A be $n \times n$ matrix generated by Rump's algorithm [12]. We choose $n = 100$ and $b = (1, 1, \cdots, 1)^T \in \mathbb{F}^n$. In this example, $1.04 \times 10^{16} < \|A\|_\infty < 1.05 \times 10^{16}$, $\|b\|_\infty = 1$ and $1.74 \times 10^{107} < \kappa(A) < 1.75 \times 10^{107}$.

In this case, a good approximate inverse can be constructed with $k = 8$ such that

$$\|RA - I\|_\infty < \alpha = 1.86 \times 10^{-4}, \tag{33}$$

where

$$R = R_1 + R_2 + \cdots + R_8 \tag{34}$$

with suitable $R_1, R_2, \cdots, R_8 \in \mathbb{F}$. The iterative refinement algorithm (20) converges with 3 iterations.

Table 2: Rump's matrix (n=100)

i	$\|v^* - v_i\|_\infty / \|v^*\|_\infty$	$\eta(v_i)$
0	7.51×10^{-6}	3.98×10^{-14}
1	5.98×10^{-11}	5.61×10^{-19}
2	4.88×10^{-16}	2.46×10^{-19}
3	3.18×10^{-19}	6.58×10^{-19}
4	3.18×10^{-19}	6.58×10^{-19}

Moreover, it is seen that

$$\begin{aligned} \beta &= \mathbf{u}^8\kappa(A) < (1.12 \times 10^{-16})^8 \times 1.75 \times 10^{107} \\ &< 4.34 \times 10^{-21}. \end{aligned} \tag{35}$$

Table 3 shows the relative errors and the backward errors of approximate solutions obtained by the iterative refinement calculations (20). The calculations are done by the same computational environment as that for the previous example.

4.3. Rump's Matrix (n=300)

Let A be $n \times n$ matrix generated by Rump's algorithm [12]. We choose $n = 300$ and $b = (1, 1, \cdots, 1)^T \in \mathbb{F}^n$. In this example, $3.10 \times 10^8 < \|A\|_\infty < 3.11 \times 10^8$, $\|b\|_\infty = 1$ and $6.28 \times 10^{59} < \kappa(A) < 6.29 \times 10^{59}$.

In this case, a good approximate inverse can be constructed with $k = 5$ such that

$$\|RA - I\|_\infty < \alpha = 1.16 \times 10^{-9}, \tag{36}$$

where

$$R = R_1 + R_2 + \cdots + R_5 \tag{37}$$

with suitable $R_1, R_2, \cdots, R_5 \in \mathbb{F}$. The iterative refinement algorithm converges (20) with 1 iteration.

Moreover, it is seen that

$$\begin{aligned} \beta &= \mathbf{u}^5\kappa(A) < (1.12 \times 10^{-16})^5 \times 6.29 \times 10^{59} \\ &< 1.11 \times 10^{-20}. \end{aligned} \tag{38}$$

Table 3 shows the relative errors and the backward errors of approximate solutions obtained by the iterative refinement calculations (20). The calculations are done by the same computational environment as that for the previous example.

Table 3: Rump's matrix (n=300)

i	$\|v^* - v_i\|_\infty / \|v^*\|_\infty$	$\eta(v_i)$
0	8.02×10^{-12}	1.42×10^{-17}
1	8.10×10^{-23}	4.07×10^{-19}
2	8.10×10^{-23}	4.07×10^{-19}

References

[1] G.B.Moler, "iterative refinement in floating point", J. Assoc. Comput. Mach., **14** 316-321 (1967)

[2] R. D. Skeel, "Iterative refinement implies numerical stability for Gaussian elimination", Math. Comp., **35** 817-832 (1980)

[3] N.J.Higham, "Iterative refinement for linear systems and LAPACK", IMA J. Numer. Anal. **17** 495-509 (1997).

[4] M.Jankowsky and H.Woznlakowski, "Iterative refinement implies numerical stability", BIT **17** 303-311 (1997).

[5] F.Tisseur, "Newton's method in floating point arithmetic and iterative refinement of generalized eigenvalue problems", SIAM J. Matrix Anal. Appl. **22** No.4 1038-1057 (2001).

[6] S. Oishi, K. Tanabe, T.Ogita, and S.M. Rump, "Convergence of Rump's method for inverting arbitrary ill-conditioned matrices", *J. Comp. and Appl. Math*, **205** 533-544 (2007).

[7] T. Ogita, S. M. Rump and S. Oishi: "Accurate Sum and Dot Product", SIAM Journal on Scientific Computing, 26/6,1955-1988,(2005)

[8] S.M.Rump:"Approximate inverses of almost singular matrices still contain useful information", Technical Report 90.1, Faculty of Information and Communication Sciences, Hamburg University of Technology (1990).

[9] S.M. Rump, T. Ogita, and S. Oishi, "Accurate Floating-point Summation I: Faithful Rounding", accepted for publication in SIAM Journal on Scientific Computing. Preprint is available from

`http://www.ti3.tu-harburg.de /publications/rump.`

[10] S.M. Rump, T. Ogita, and S. Oishi, "Accurate Floating-point Summation II: Sign, K-fold Faithful and Rounding to Nearest", submitted for publication in SIAM Journal on Scientific Computing. Preprint is available from

`http://www.ti3.tu-harburg.de /publications/rump.`

[11] T. Ohta, T. Ogita, S. M. Rump and S. Oishi: "A Method of Verified Numerical Computation for Ill-conditioned Linear System of Equations", Journal of JSIAM,15:3 (2005), pp. 269–287 in Japanese.

[12] S. M. Rump: A class of arbitrarily ill-conditioned floating-point matrices, SIAM J. Matrix Anal. Appl., 12:4 (1991), 645–653.

[13] J. D. Rigal and J. Gaches:"On the compativility of a given solution with the data of a linear equation", J. Assoc. Comput. Mach., 14 (1967), 543-548.