GPUを用いた線形方程式の精度保証法

中村 祐太郎
早稲田大学 基幹理工学部 応用数理学科
e-mail : n.yutaro@fuji.waseda.jp

1 はじめに

線形方程式 $AX = B$ を最近点丸めの事前誤差評価を用いて評価を行い，GPU 上での実装を可能にする．$\mathbb{F}^{n\times n}$ を浮動小数点数の集合とする．本報告における計算は IEEE754 に基づく倍精度精度浮動小数点演算を用い [1]，アンダーフロー，オーバーフローは起こらないと仮定する．Oishi-Rump が文献 [2] において，IEEE754 に基づく丸めモードの変更を用い，連立 1 次方程式 $Ax = b, A \in \mathbb{F}^{n\times n}, b \in \mathbb{F}^n$ における解を精度保証付きで求める方法を提案している．これは，線形方程式にも拡張が可能である．

線形方程式 $AX = B$ において区間を持った B を以下のように表す．($A \in \mathbb{F}^{n\times n}$ は区間を持たない)

$$B = < \mathrm{mid}_B, \mathrm{rad}_B > \tag{1}$$

このとき，$\mathrm{mid}_B \in \mathbb{F}^{n\times n}$ は B の中心.$\mathrm{rad}_B \in \mathbb{F}^{n\times n}$ は B の半径を表す．$\tilde{X} \in \mathbb{F}^{n\times n}$ を近似解．$R \in \mathbb{F}^{n\times n}$ としたとき，$\|RA - I\|_\infty \le 1$ であれば，A は正則であり，$AX = B$ における精度保証式は，区間を持った線形方程式の場合も同様に

$$\|X - \tilde{X}\|_\infty \le \frac{\|R(A\tilde{X} - B)\|_\infty}{1 - \|RA - I\|_\infty} \tag{2}$$

と評価される．

2 先行研究

連立 1 次方程式 $Ax = b$ においては，以下の評価式が成立する．

$$\|x - \tilde{x}\|_\infty \le \frac{\|R(A\tilde{x} - B)\|_\infty}{1 - \|RA - I\|_\infty}. \tag{3}$$

$Ax = b$ における上記した精度保証式を評価するために,$\|RA - I\|_\infty \le \alpha$ と $\|R(A\tilde{x} - B)\|_\infty \le \beta$ を丸めの向きを変更する手法で求めるアルゴリズムを以下に記す．本報告中のアルゴリズムはコード記述の簡略化のために Matlab，INTLAB における関数に似たコードで記述する．

Algorithm 1 (Oishi and Rump [2]). $A \in \mathbb{F}^{n\times n}$, $b \in \mathbb{F}^{n\times n}$, $\tilde{x}$ を $Ax = b$ の近似解，R を A の近似逆行列としたとき，丸めの向きを変更して $\|RA - I\|_\infty$ の上限 α と $\|R(A\tilde{X} - B)\|_\infty$ の上限 β を求めるアルゴリズム

function$[\alpha, \beta] = \mathrm{std.rnd}(A, b, \tilde{x}, R)$
　setround(-1)
　$\underline{G} = \mathrm{fl}(RA - I)$;
　$\underline{r} = \mathrm{fl}(A\tilde{x} - b)$;
　setround$(+1)$
　$\bar{G} = \mathrm{fl}(RA - I)$;
　$\underline{r} = \mathrm{fl}(A\tilde{x} - b)$;
　$\alpha = \mathrm{fl}(\|\max(|\underline{G}|, |\bar{G}|)\|_\infty)$;
　$r_{\mathrm{mid}} = \dfrac{(\underline{r} + \bar{r})}{2}$;
　$r_{\mathrm{rad}} = r_{\mathrm{mid}} - \underline{r}$;
　setround(-1)
　$\underline{t} = \mathrm{fl}(R \cdot r_{\mathrm{mid}})$;
　setround$(+1)$
　$\bar{t} = \mathrm{fl}(R \cdot r_{\mathrm{mid}})$;
　$\beta = \mathrm{fl}(\|\max(|\underline{t}|, |\bar{t}|) + |R| \cdot r_{\mathrm{rad}}\|_\infty)$;

Ogita-Rump-Oishi が (3) の関係式において，最近点丸めにおける浮動小数点演算の誤差評価を用いて，丸めモードの変更を用いない手法を提案している [3]．$\mathrm{fl}(\cdot)$ は括弧内の計算をすべて最近点丸め方向に計算を行ったことを表す．このとき，(3) の関係式は最近点丸めにおける計算のみを用いて，以下のように計算される．

$$\|x - \tilde{x}\|_\infty \le \mathrm{fl}\left(\frac{\beta/(1-\alpha)}{1 - 3\mathbf{u}}\right). \tag{4}$$

$$\mathbf{u} = 2^{-53},\ e = (1, \ldots, 1)^T \in \mathbb{F}^n$$
$$\gamma_n = \mathrm{fl}\frac{n\mathbf{u}}{1 - n\mathbf{u}},\ \tilde{\gamma_n} = \mathrm{fl}\left(\frac{n\mathbf{u}}{1 - n\mathbf{u}}\right)$$

と定義する．本報告では，アンダーフローが起こらないと仮定するため，文献 [3] における誤

差評価式を用い精度保証式 (3) における α を再評価する.

$$M = \mathrm{fl}(RA) \tag{5}$$

とすると

$$\|RA - I\|_\infty \le \|M - I\|_\infty + \|\gamma_n |R||A|\|_\infty \tag{6}$$

このとき

$$\begin{aligned} \|\gamma_n|R||A|\|_\infty &= \|\gamma_n|R|(|A|e)\|_\infty \\ &\le \|\gamma_n|R|(1+\mathbf{u})^{n-1}\mathrm{fl}(|A|e)\|_\infty \\ &\le \|\gamma_n(1+\mathbf{u})^{2n-1}\mathrm{fl}(|R||A|e)\|_\infty \\ &\le \mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+1}\||R|(|A|e)\|_\infty}{1-2n\mathbf{u}}\Big) \\ &=: c_1 \end{aligned} \tag{7}$$

次に

$$\begin{aligned} |\mathrm{fl}(M-I) - (M-I)| &\le \mathbf{u}\cdot I \\ &\le \mathrm{fl}(\mathbf{u}\cdot I) \end{aligned} \tag{8}$$

よって,

$$\|M - I\|_\infty \le \|\mathrm{fl}(M-I)\|_\infty + \|\mathrm{fl}(\mathbf{u}\cdot I)\|_\infty \tag{9}$$

同様に最近点丸めを用いてノルムの評価を行うと,

$$\begin{aligned} \|\mathrm{fl}(M-I)\|_\infty &\le \mathrm{fl}\Big(\frac{\|M-I\|_\infty}{1-n\mathbf{u}}\Big) \\ &=: c_2 \end{aligned} \tag{10}$$

と

$$\begin{aligned} \|\mathrm{fl}(\mathbf{u}\cdot I)\|_\infty &\le \mathrm{fl}(\|\mathbf{u}\cdot I\|_\infty) \\ &=: c_3 \end{aligned} \tag{11}$$

となる. よって, $\|RA - I\|_\infty$ の上限は,c_1, c_2, c_3 より

$$\begin{aligned} \|RA - I\|_\infty &\le c_1 + c_2 + c_3 \\ &\le (1+\mathbf{u})^2\mathrm{fl}(c_1+c_2+c_3) \\ &\le \mathrm{fl}\Big(\frac{c_1+c_2+c_3}{1-3\mathbf{u}}\Big) \end{aligned} \tag{12}$$

このアルゴリズムを以下に記す.

Algorithm 2 $A \in \mathbb{F}^{n\times n}$, R を A の近似逆行列としたとき, $\|RA - I\|_\infty$ の上限 α を求めるアルゴリズム

$$\begin{aligned} &\text{function}[\alpha] = \text{Alpha}(A, R) \\ &\quad c_1 = \mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+1}\||R|(|A|e)\|_\infty}{1-2n\mathbf{u}}\Big) \\ &\quad M = \mathrm{fl}(RA) \\ &\quad c_2 = \mathrm{fl}\Big(\frac{\|M-I\|_\infty}{1-n\mathbf{u}}\Big) \\ &\quad c_3 = \mathrm{fl}(\|\mathbf{u}\cdot I\|_\infty) \\ &\quad \alpha = \mathrm{fl}\Big(\frac{c_1+c_2+c_3}{1-3\mathbf{u}}\Big) \end{aligned}$$

3 本報告における手法

連立 1 次方程式における精度保証法を線形方程式 $AX = B$ に拡張し, B が区間を持つ場合において, 事前誤差評価を用いた最近点丸めのみを用いた手法を提案する. B が区間を持つとき,$\|RA - I\|_\infty \le \alpha$ と $\|R(A\tilde{X} - B)\|_\infty \le \beta$ を丸め変更する手法で求めるアルゴリズムを以下に記す.

Algorithm 3 $A \in \mathbb{F}^{n\times n}$,$\mathrm{mid}_B \in \mathbb{F}^{n\times n}$,$\tilde{X}$ を $AX = B$ の近似解,R を A の近似逆行列としたとき, 丸めを変更して $\|RA - I\|_\infty$ の上限 α と $\|R(A\tilde{X} - B)\|_\infty$ の上限 β と error を求めるアルゴリズム

$$\begin{aligned} &\text{function}[\alpha,\beta] = \text{std.rnd.int}(A, B, \tilde{X}, R) \\ &\quad \text{setround}(-1) \\ &\quad \underline{G} = \mathrm{fl}(RA - I); \\ &\quad \underline{r} = \mathrm{fl}(A\tilde{X} - \mathrm{mid}_B); \\ &\quad \text{setround}(+1) \\ &\quad \bar{G} = \mathrm{fl}(RA - I); \\ &\quad \bar{r} = \mathrm{fl}(A\tilde{X} - \mathrm{mid}_B); \\ &\quad \alpha = \mathrm{fl}(\|\max(|\underline{G}|, |\bar{G}|)\|_\infty); \\ &\quad r_{\mathrm{mid}} = \frac{(\mathrm{mid}(\underline{r}) + \mathrm{mid}(\bar{r}))}{2}; \\ &\quad r_{\mathrm{rad}} = r_{\mathrm{mid}} - \mathrm{mid}(\underline{r}) + \mathrm{rad}(\underline{r}) + \mathrm{rad}(\bar{r}); \\ &\quad \text{setround}(-1) \\ &\quad \underline{t} = \mathrm{fl}(R\cdot r_{\mathrm{mid}}); \\ &\quad \text{setround}(+1) \\ &\quad \bar{t} = \mathrm{fl}(R\cdot r_{\mathrm{mid}}); \\ &\quad \beta = \mathrm{fl}(\|\max(|\underline{t}|, |\bar{t}|) + |R|\cdot r_{\mathrm{rad}}\|_\infty); \end{aligned}$$

線形方程式 $AX = B$ における精度保証法において, A は区間なしを仮定しているため, α

は Algprithm 2 と同様である．次に $\|R(A\tilde{X}-B)\|_\infty$ の評価を行う．B は区間を持ち,$B=<\mathrm{mid}_B,\mathrm{rad}_B>$ と表されるため,

$$A\tilde{X}-B \in <\mathrm{fl}(A\tilde{X}-\mathrm{mid}_B),\mathrm{rad}_B+\gamma_{n+1}(|A||\tilde{X}|+|\mathrm{rad}_B|)>$$

となる．ここで,

$$\begin{aligned}\mathrm{Mid}&=\mathrm{fl}(A\tilde{X}-B)\\ \mathrm{Rad}&=\mathrm{rad}_B+\gamma_{n+1}(|A||\tilde{X}|+|B|)\end{aligned}$$

とおくと,

$$\begin{aligned}&\|R(A\tilde{X}-B)\|_\infty\\ &\le \||R\cdot\mathrm{Mid}|+|R|\mathrm{Rad}\|_\infty\\ &\le \Big\||R\cdot\mathrm{Mid}|+|R|\mathrm{rad}_B\\ &+\gamma_{n+1}(|R||A||\tilde{X}|+|R||\mathrm{mid}_B|)\Big\|_\infty\\ &\le \Big\||R\cdot\mathrm{Mid}|+|R|\cdot\mathrm{rad}_B+\gamma_{n+1}(|R||A||\tilde{X}|\\ &+|R||\mathrm{mid}_B|)+\gamma_n|R||\mathrm{Mid}|\Big\|_\infty\\ &=\Big\||R\cdot\mathrm{Mid}|e+|R|\mathrm{rad}_B\cdot e\\ &+\gamma_{n+1}(|R||A||\tilde{X}|e+|R||\mathrm{mid}_B|e)\\ &+\gamma_n|R||\mathrm{Mid}|e\Big\|_\infty \qquad (13)\end{aligned}$$

このとき

$$\begin{aligned}&|\mathrm{fl}(R\cdot\mathrm{Mid})|\\ &\le(1+\mathbf{u})^{n-1}\mathrm{fl}(|\mathrm{fl}(R\cdot\mathrm{Mid})|e)\\ &\le\mathrm{fl}\Big(\frac{(|R\cdot\mathrm{Mid}|)e}{1-n\mathbf{u}}\Big)\\ &=:s_1 \qquad (14)\end{aligned}$$

次に

$$\begin{aligned}&|R|\mathrm{rad}_B\cdot e\\ &\le|R|(1+\mathbf{u})^{n-1}\mathrm{fl}(\mathrm{rad}_B\cdot e)\\ &\le(1+\mathbf{u})^{2n-1}\mathrm{fl}(|R|(\mathrm{rad}_B\cdot e))\\ &\le\mathrm{fl}\Big(\frac{(|R|\cdot(\mathrm{rad}_B\cdot e)}{1-2n\mathbf{u}}\Big)\\ &=:s_2 \qquad (15)\end{aligned}$$

また

$$\begin{aligned}&\gamma_{n+1}(|R||A||\tilde{X}|e)\\ &\le\gamma_{n+1}|R|(1+\mathbf{u})^{2n-1}\mathrm{fl}(|A|(|\tilde{X}|e))\\ &\le\gamma_{n+1}(1+\mathbf{u})^{3n-1}\mathrm{fl}(|R|(|A|(|\tilde{X}|e)))\\ &\le\mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+2}|R|(|A|(|\tilde{X}|e))}{1-3n\mathbf{u}}\Big)\\ &=:s_3 \qquad (16)\end{aligned}$$

次に

$$\begin{aligned}&\gamma_{n+1}|R||\mathrm{mid}_B|e\\ &\le\mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+2}(|R|(|\mathrm{mid}_B|e))}{1-2n\mathbf{u}}\Big)\\ &=:s_4 \qquad (17)\end{aligned}$$

最後に

$$\begin{aligned}&\gamma_n|R||\mathrm{Mid}|e\\ &\le\mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+1}(|R|(|\mathrm{Mid}|e))}{1-2n\mathbf{u}}\Big)\\ &=:s_5 \qquad (18)\end{aligned}$$

ここで,$\|R(A\tilde{X}-B)\|_\infty$ の評価を行う．

$$\begin{aligned}&\|R(A\tilde{X}-B)\|_\infty\\ &\le\|s_1+s_2+s_3+s_4+s_5\|_\infty\\ &\le\|((1+\mathbf{u})^4\mathrm{fl}s_1+s_2+s_3+s_4+s_5)\|_\infty\\ &\le\mathrm{fl}\Big(\frac{\|s_1+s_2+s_3+s_4+s_5\|_\infty}{(1-5\mathbf{u})}\Big) \qquad (19)\end{aligned}$$

このアルゴリズムを以下に記す．

Algorithm 4 $A\in\mathbb{F}^{n\times n}$,$\mathrm{mid}_B,\mathrm{rad}_B\in\mathbb{F}^{n\times n}$,$\tilde{X}$ を $AX=B$ の近似解,R を A の近似逆行列としたとき,$\|R(A\tilde{X}-B)\|_\infty$ の上限 β を求めるアルゴリズム

$$\begin{aligned}&\mathrm{function}[\beta]=\mathrm{Betaint}(A,B,\tilde{X},R)\\ &\quad\mathrm{Mid}=\mathrm{fl}(A\tilde{X}-B)\\ &\quad\mathrm{Rad}=\mathrm{rad}_B+\gamma_{n+1}(|A||\tilde{X}|+|B|)\\ &\quad s_1=\mathrm{fl}\Big(\frac{(|R\cdot\mathrm{Mid}|)e}{1-n\mathbf{u}}\Big)\\ &\quad s_2=\mathrm{fl}\Big(\frac{(|R|\cdot(\mathrm{rad}_B\cdot e)}{1-2n\mathbf{u}}\Big)\\ &\quad s_3=\mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+2}|R|(|A|(|\tilde{X}|e))}{1-3n\mathbf{u}}\Big)\\ &\quad s_4=\mathrm{fl}\Big(\frac{|R||\mathrm{rad_B}|}{1-(n+1)\mathbf{u}}\Big)\\ &\quad s_5=\mathrm{fl}\Big(\frac{\tilde{\gamma}_{n+2}(|R|(|\mathrm{mid}_B|e))}{1-2n\mathbf{u}}\Big)\\ &\quad\beta=\mathrm{fl}\Big(\frac{\|s_1+s_2+s_3+s_4+s_5\|_\infty}{(1-5\mathbf{u})}\Big)\end{aligned}$$

4 数値実験

それぞれのアルゴリズムを実装し数値実験を行った．各項目において，異なる行列で 10 回計算を行い，その平均値を表に表す．表における”-”は $\alpha\ge1$ となった場合であり，精度保証失敗を表す．

- 計算環境

1. CPU: Intel(R) Xeon(R) CPU X5680 @3.33GHz (ハイパースレッディング有/24コア相当)
2. Memory: 96 GB
3. OS: Yellow dog Linux for CUDA 6.1
4. GPU: NVIDIA TeslaC2070
5. MATLAB (R2012a, Parallel computing tool box), GotoBLAS2 1.13

- テスト行列の作成方法

randn('state',i)
A=randn(n)
A=gallery($'randsvd', n, 10^s, 3, n, n, 1$)
B=A*ones(n)
$B\_ = \mathrm{midrad}(B, \mathrm{u} * \mathrm{abs}(B))$
$\mathrm{mid_B}$=B
$\mathrm{rad_B} = \mathrm{u} \cdot |\mathrm{mid_B}|$
R=inv(A)
$\tilde{X}$=A\ B

表 1. 各手法による α の精度保証結果の比較

n	std.rnd.int	std_int	std_int GPU
100	3.00e-12	2.99e-10	2.99e-10
500	6.66e-10	8.83e-09	8.81e-09
1000	1.42e-10	2.63e-08	2.63e-08
2000	3.68e-10	9.43e-08	9.44e-08
5000	9.23e-09	3.99e-06	3.99e-06
10000	1.65e-08	1.06e-05	1.06e-05

表 2. 各手法による $\|x - \tilde{x}\|_\infty$ の上限の結果の比較

n	std.rnd.int	std_int	std_int GPU
100	1.22e-10	3.32e-08	3.32e-08
500	2.15e-09	4.59e-06	4.59e-06
1000	6.10e-09	2.70e-05	2.70e-05
2000	1.47e-09	1.92e-04	1.92e-04
5000	7.53e-04	2.02e-02	2.02e-02
10000	1.10e-06	1.07e-01	1.07e-01

5 まとめ

本報告では，B が区間を持つような線形方程式 $AX = B$ における最近点丸めのみを用いた精度保証法を提案した．また，最近点丸めを用

表 3. 各手法の計算時間 (秒) の比較

n	std.rnd.int	std_int	std_int GPU
100	0.018	0.0103	0.0281
500	0.105	0.096	0.086
1000	0.432	0.293	0.233
2000	4.084	1.896	1.018
5000	29.69	22.23	8.463
10000	261.8	159.6	52.87

表 4. 条件数を変えた場合における α の上限の比較 (n=1000)

$\mathrm{cond}_2(A)$	std.rnd.int	std_int	std_int GPU
10^2	3.60e-12	8.90e-10	9.01e-10
10^4	1.07e-10	4.77e-08	4.84e-08
10^6	5.25e-09	3.33e-06	3.39e-06
10^8	3.30e-7	2.59e-04	2.64e-04
10^{10}	2.27e-05	2.20e-02	2.20e-02
10^{12}	1.80e-03	-	-

表 5. 条件数を変えた場合における $\|x - \tilde{x}\|_\infty$ の上限の比較 (n=1000)

$\mathrm{cond}_2(A)$	std.rnd.int	std_int	std_int GPU
10^2	2.21e-10	9.15e-07	9.15e-07
10^4	7.44e-09	4.91e-05	4.91e-05
10^6	4,42e-07	3.4e-03	3.4e-03
10^8	3.15e-05	2.68e-01	2.68e-01
10^{10}	2.5e-03	22.7	22.77
10^{12}	2.13e-01	-	-

いたため，GPU への実装可能性も同時に示した．

提案手法を用いると丸めの変更を行った場合に比べ，計算時間は改善されるが精度が悪くなる．また GPU を用いて計算を行うと n が大きくなると, 計算速度が上がるという結果が得られた. 提案手法では, 条件数を 10^{12} 以上にすると精度保証に失敗するという結果が得られた．これは, 事前誤差評価を用いたために起こる過大評価である．

参考文献

[1] ANSI/IEEE: IEEE Standard for Binary Floating-Point Arithmetic: ANSI/IEEE Std 754-1985, New York, IEEE, 1985.

[2] S. Oishi, S. M. Rump: Fast Verification Solutions of Matrix Eqiations, Numer. Math., 90:4 (2002), 755-773.

[3] T. Ogita, S.M. Rump, S. Oishi, Verified solution of linear systems without directed rounding, Technical Report, No. 2005-04, Advanced Research Institute for Science and Engineering, Waseda University.

[4] 大石 進一: 精度保証付き数値計算, コロナ社,2000.