計算工学講演会論文集　Vol.30 (2025年6月)　日本計算工学会

ILU (0) 前処理付き BiCGSTAB 法における低精度演算導入の分析

The analysis of the introduction of low-precision arithmetic in the ILU (0) preconditioned BiCGSTAB method

久木田仁[1)]，藤井昭宏[1)]，田中輝雄[1)]
Jin Kukita, Akihiro Fujii and Teruo Tanaka

1) 工学院大学大学院工学研究科(〒163-8677 東京都新宿区西新宿1-24-2)

Efforts to accelerate numerical computations using specialized arithmetic units, such as GPU and low-precision matrix multiplication units for AI applications, have been extensively conducted. With the increasing prevalence of such low-precision arithmetic units and their widespread application in numerical computations, research on mixed-precision algorithms that balance accuracy and computational speed has become essential. In this study, we analyze the effect of aggressive mantissa bit reduction in the ILU (0) preconditioned BiCGSTAB method.

Key Words : *sparse linear solver, low precision computing, BiCGSTAB method*

1.　はじめに

GPU や AI 向け低精度行列積演算器などの特殊な演算器は今や高精度な演算が求められる数値計算においても無視できない存在となっている．これらの演算器は精度の低い浮動小数点形式を用いることが特徴である．このような低精度演算器の普及と，これの数値計算への応用の浸透に伴い，精度と速度の両立を図る混合精度アルゴリズムの研究と理解が求められている．深谷らは ILU (0) 前処理付き GMRES (m) 法において仮数部ビットの削減により低精度計算導入時の影響を分析し，低精度計算導入の有効性を示した[1]．ILU (0) 前処理付き GMRES (m) 法ではリスタート毎に残差を再計算するため収束判定後の解の誤差が許容範囲内であることが保証されるが，反復内の精度が低下すると反復回数が増加した．この精度低下と反復回数増加のトレードオフを踏まえ，積極的な低精度計算の導入が高速化において有用である可能性があると結論づけられていた．

また，Zhao らによって BiCGSTAB 法に低精度演算を取り入れる MP-IR 方式が提案されている[2]．BiCGSTAB 法を単純に低精度化した場合，その特性上解の精度が低くなってしまう．この問題に対し，MP-IR 方式では反復改良法の形をとり収束判定後の精度の保証が行われた．

本研究では，ILU (0) 前処理付き BiCGSTAB 法に 52bit 中 40bit といった積極的な仮数部ビット削減を行った影響を数値実験により分析する．実験はテスト行列を用いて仮数部の下位ビットを削除することで仮想的に低精度化を行う．低精度化の対象は前処理行列，または BiCGSTAB 処理内部のデータである．BiCGSTAB 処理内部のデータの低精度化については MP-IR 方式を適用した場合の評価も行う．

2.　BiCGSTAB法

(1) BiCGSTAB法

BiCGSTAB 法は，大規模で疎かつ非対称な行列を係数とする連立一次方程式に対する Krylov 部分空間法の一種であり，広く利用されている．同様の係数行列を対象とする Krylov 部分空間法である GMRES 法と比較すると以下のような特徴をもつ．

- 高速な場合がある
- メモリ使用量が一定である
- 誤差累積の影響を受けやすい
- 反復改良の構造を有していない

(2) ILU (0) 前処理付きBiCGSTAB法

BiCGSTAB 法には前処理を用いる場合があり，その代表的な手法のひとつが ILU (0) 前処理である．これは不完全 LU 分解により係数行列 A の非ゼロ要素の位置を維持したまま下三角行列 L と上三角行列 U の積の形へ近似的に分解し，これらを前処理に用いる．この前処理により，収束の加速や悪条件な問題の求解をしやすくする．

(3) 混合精度ILU (0) 前処理付きBiCGSTAB法

BiCGSTAB 法は，反復処理にて更新した残差を用いて収束判定を行うアルゴリズムである．そのため，単純に低精度演算を導入すると誤差が累積し，真の残差と算出された残差に差が生じてしまう．この差により，収束判定後の解が許容範囲外の誤差をもつことがある．この問題を解決する混合精度アルゴリズムに，MP-IR (Mixed Precision – Iterative Refinement) 方式がある．これは反復改良法の残差更新に低精度の解法を用いることで，収束判定の精度を保証しつつ，低精度演算の導入による高速化を図る手法である．本研究では，これを用いて MP-IR 方式の ILU (0) 前処理付き BiCGSTAB 法を扱う．

3.　データの低精度化

(1) 浮動小数点形式

浮動小数点形式はコンピュータで数値を表す際に広く用いられている形式である．これは符号部，指数部，仮数部の 3 つのデータで構成され，それぞれ符号部は正負，指数部はスケール, 仮数部は詳細な値を表している．指数部のデータ量が少ないほど表せる数の範囲が狭くなり，仮数部のデータ量が少ないほど表せる数の精度が低くなる. 広く用いられる形式である倍精度 (FP64) では指数部が 11bit，仮数部が 52bit であり，単精度 (FP32) では指数部が 8bit，仮数部が 23bit である.

(2) 浮動小数点形式データの低精度化

本研究では，対象となるデータの仮数部の下位ビットを削除することで擬似的にデータの低精度化を行う．具体的には，FP64 のデータに対して下位最大 52bit を強制的に 0 にするビット演算を行うことで低精度化し，実際の演算は FP64 で行う. 本実験に用いたプログラムは C++ 言語を用いて図 1 の関数として実装した．この関数は double 型配列の開始位置ポインタ var と, 削除するビット数 d，配列の長さ n を引数にとる．この関数は以下のような処理を行う.

i. 64bit 整数型変数 mask=1 として宣言
ii. mask を d の値だけ左へビットシフト
iii. mask から 1 を減じる
iv. ビット反転演算子 (~) により mask のビットを反転
v. 作成したマスクと元データで論理和 (&) をとる
vi. 配列内の全要素に対して i - ivの操作を繰り返す

i – ivでは，下位 d ビットが 0 のマスクを作成する．v では，元データの下位 d ビットを 0 にし，上位ビットの値を保持する操作を行っている.

(3) 低精度化の対象

本研究では，前処理行列と BiCGSTAB 処理内部のデータに対して擬似的な低精度化を行う．前処理行列に関しては，ILU (0) 分解処理後の行列 L, U に対して低精度化を行い，LU (0) 分解処理中の低精度化や前処理部分の低精度化は行わない．BiCGSTAB 処理内部のデータに関しては，前処理付き BiCGSTAB 法のアルゴリズムにおける演算子の処理が完了するごとに低精度化を行っている．例として, 内積では, 低精度化されたベクトル同士に対して FP64 精度の内積演算をし, その計算結果であるスカラー値に対して低精度化を行う.

```
void clearLowerBitsVector(double *var, int d, int n) {
    uint64_t mask = 1;
    for (int i = 0; i < n; i++) {
        uint64_t *ptr = reinterpret_cast<uint64_t *>(&var[i]);
        *ptr &= ~static_cast<uint64_t>((mask << d) - 1);
    }
}
```

図-1 低精度化に用いた関数

4.　数値実験

(1) 実験目的

ILU (0) 前処理付き BiCGSTAB 法の振る舞いを，テスト行列を用いた数値実験により分析する. 残差履歴, 収束に要した反復回数，最終的に求まった解の真の残差を仮数部 bit 数ごとに取得する．なお，実験は以下の 3 つのパターンに分けて行う.

a) 前処理行列を低精度化，BiCGSTAB は通常精度
b) BiCGSTAB を低精度化，前処理行列は通常精度
c) b) に加え，MP-IR 方式を導入

(2) 実験条件

本項では，数値実験に用いたテスト行列や収束条件の設定について述べる. 本実験では, 先行研究を参考に問題設定を行った.

係数行列となるテスト行列には, 2 次元移流拡散方程式を離散化して得られる行列サイズ 3600 の疎な五重対角行列を用いた. この行列は $k^{(H)}$というパラメータによって一部の対角成分の値を指定することができる．実験 a) にて $k^{(H)}=1$，$k^{(H)}=10^5$ の場合を扱うが，$k^{(H)}=10^5$ の方が大きい条件数をもち，悪条件となる.

BiCGSTAB 法の収束条件には，相対残差が一定以下となることと，反復回数が上限に達することの 2 つを設定した．具体的には，相対残差が 10^{-14} 以下となる，反復回数が行列サイズの 3 倍である 10800 を超える，という条件で処理が終了する．また，MP-IR 方式では反復改良法の構造を有する外部反復と，BiCGSTAB 法による内部反復が存在する．これらに対し，外部反復 1 回あたりの内部反復数は 500 回，総反復数は 10800 回を上限とした．内部反復中に総反復数の上限に到達した場合はその時点で処理を終了する.

(3) 実験結果

a) 前処理行列

前処理行列に対して仮数部ビット削減を行った実験結果を示す. $k^{(H)}=1$ の場合の相対残差ノルム履歴を図 2 に示す．横軸は反復回数，縦軸は相対残差 2 ノルムである．各折れ線は仮数部ビット数ごとの相対残差ノルム履歴を表し，特徴的なパターンについては下部に何 bit 分の精度を保持しているかを表している．以後，x bit の精度を保持するパターンを“x bit”と表記する．収束までの反復回数を比較すると，最も少ない反復回数で収束したのは”11 bit”であった．また，”1 bit“では収束が顕著に遅くなり，“0 bit“では一度残差が大きくなった後に見かけ上収束した．求まった解の精度を比較すると，“0 bit“以外では約 10^{-13} の残差であったが，“0 bit“では 10^{-3} 程度の残差となった．すなわち, “0 bit“では解を正しく求められなかった. なお, 前処理を行わない場合は 10800 回の反復上限に達して打ち切られ，求まった解の真の残差は 10^{-13} 程度であった.

$k^{(H)}=10^5$ の場合の相対残差ノルム履歴を図 3 に示す．収束までの反復回数を比較すると，最も少ない反復回数で

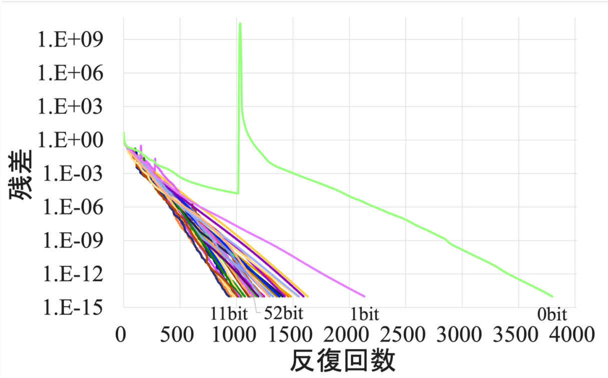

図-2 前処理行列に対する削減時の残差履歴（$k^{(H)}=1$）

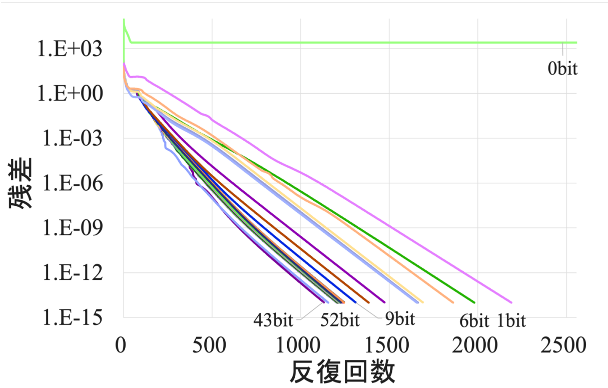

図-3 前処理行列に対する削減時の残差履歴（$k^{(H)}=10^5$）

収束したのは"43 bit"であった．また，"9 bit"以下では反復回数が増加し，"0 bit"では 10^3 付近に停滞して収束判定に至らなかった．求まった解の精度を比較すると，"2 bit"以上では 10^{-13}，"1 bit"では 10^{-12}，"0 bit"では 10^4 程度の残差であった．なお，前処理を行わない場合は求解できなかった．

b) BiCGSTAB処理内部

ILU(0)前処理付き BiCGSTAB 法の BiCGSTAB 処理内部で扱うデータに対して仮数部ビット削減を行った実験結果を示す．$k^{(H)}=1$ の場合の相対残差ノルム履歴を図 4 に示す．BiCGSTAB 法の各反復で得られた x の値から取得した真の残差の履歴を図 5 に示す．また，各仮数部ビット収束判定後の真の残差を図 6 に示す．図 6 の横軸は保持した仮数部ビット数を表し，縦軸は収束判定後の解をもとに再計算した真の相対残差である．図 4 について，収束に要した反復回数では"0 bit"が最も少なく，"30 bit"が最も多かった．図 6 の収束判定後の真の残差は，ビット数が少ないほど大きくなった．すなわち，仮数部ビット削減を BiCGSTAB 法へと単純に取り入れると解の精度が落ちた．図 5 の真の残差履歴では，仮数部ビット数ごとに一定の残差に至ると更新が止まるという特性がみられた．この結果から，先述した更新の停止による真の残差の改善を伴わない反復回数の増加の存在を確認した．

c) MP-IR方式におけるBiCGSTAB処理内部

前節の実験結果では，仮数部ビット削減を BiCGSTAB 法へと単純に取り入れると解の精度が落ちることが示さ

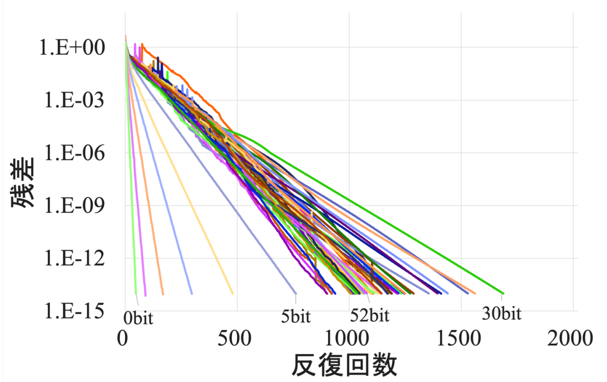

図-4 BiCGSTAB 処理内部に対する削減時の残差履歴

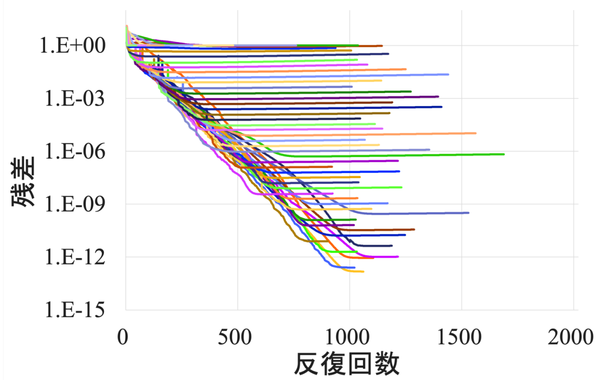

図-5 BiCGSTAB処理内部に対する削減時の真の残差履歴

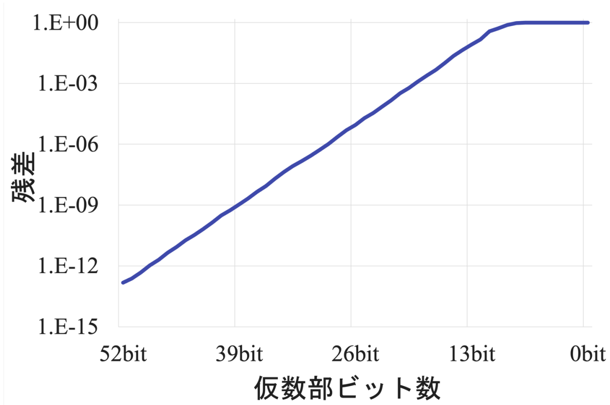

図-6 BiCGSTAB処理内部に対する削減時の最終残差

れた．本節では，真の残差を用いて収束判定を行う MP-IR 方式を用いた数値実験の評価を行う．図 7 に仮数部ビット数ごとの総反復回数を示す．本実験の条件では，総反復回数の制限が 10800 であり，これを超えると処理は中断される．そのため，反復回数が 10800 となった"16 bit"以下では解の精度が保証されない．図 8 に仮数部ビット数ごとの最終的な残差を示す．仮数部ビット数が"17 bit"以上の場合の残差は 10^{-15} 程度であり，"16 bit"以下では残差が増加していることがわかる．これは反復回数から予想された内容と一致している．なお，"0 bit"付近では相対残差が 1 となったが，x が 0 ベクトルになってしまっているためと考えられる．

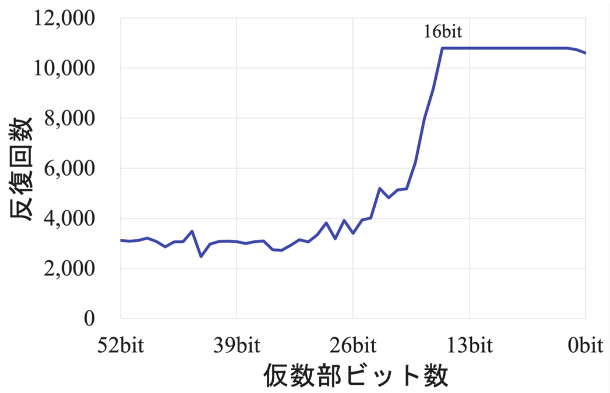

図-7 MP-IR方式での仮数部ビット数毎の反復回数

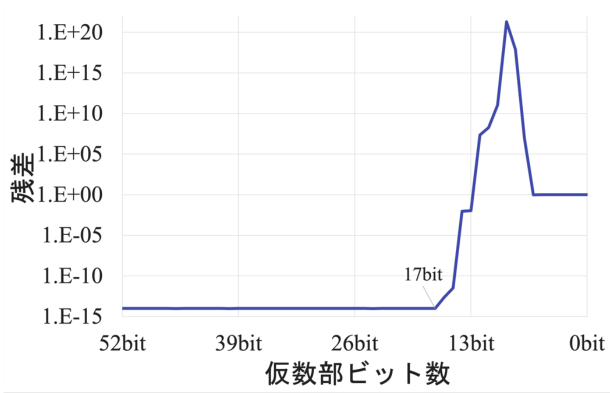

図-8 MP-IR方式での仮数部ビット数毎の最終残差

5. 考察

本研究では，ILU (0) 前処理付き BiCGSTAB 法に対する仮数部ビット削減の影響をテスト行列により分析することを目的として数値実験を行った．

前処理行列に対する大幅なビット削減はテスト行列に対し有効であった．前処理行列に対してビット削減を行った場合，$k^{(H)}=1$の”2 bit”以上，$k^{(H)}=10^5$の”10 bit”以上の範囲では顕著な影響が見られなかった．すなわち，52bit中40bitといった大幅な削減を行っても反復回数や求解精度には影響がなかった．一方で，$k^{(H)}=1$のテスト行列では”2 bit”程度までは大きな影響なく求解できたものの，悪条件である$k^{(H)}=10^5$のテスト行列に対しては”9 bit”以下の範囲で収束性の悪化がみられた．このことから，ビット削減の許容範囲は悪条件の行列ほど狭まる可能性がある．

BiCGSTAB 処理内部のデータに対する仮数部ビット削減は，解の精度の低下に直結した．また，これを改善する MP-IR 方式はテスト行列に対し有効な側面をもった．BiCGSTAB 処理内部のデータに対して仮数部ビット削減を行うと，削減数に応じて解の精度が低下した．ここで，解の精度を保証するアルゴリズムである MP-IR 方式を用いると，”17 bit”以上であれば精度を保った求解が可能になった．ビット削減なしの一般的な解法では反復回数が約 1100，解の残差が 10^{-13} であったのに対し，MP-IR 法の”17 bit”以上では反復回数が約 2400 から 8000，解の残差は 10^{-15} であった．先行研究と同様に，MP-IR 方式には，解の残差が小さくなり BiCGSTAB 法に低精度演算を導入できるというメリットがある反面，反復回数が増加するデメリットもあった．

6. おわりに

本研究では，大規模で疎かつ非対称な係数行列をもつ連立一次方程式に対する解法である ILU (0) 前処理付き BiCGSTAB 法における積極的な低精度化の分析を目的とした．ビット演算を用いた仮数部ビット削減により低精度化を仮想的に再現し，テスト行列を用いた数値実験を行った．前処理行列に対しては仮数部ビット数を 52bit から 11bit まで削減する大幅なビット削減を行っても反復回数や解の精度への影響は軽微だった．また，BiCGSTAB 処理内部のデータに仮数部ビット削減を行うと削減数に応じて解の精度が低下した．これに対し，MP-IR 方式の導入により，仮数部 17bit 以上において解の精度を落とさない求解が可能となった．反復回数に着目すると，ビット削減なしの一般的な解法では 1100, MP-IR 方式では 2400 から 8000 へと増加するが，解の残差は 10^{-13} から 10^{-15} へ改善され，データや演算の低精度化が可能となり，MP-IR 方式の有効性が示唆された．

今後の課題として，まず低精度化実装の詳細化が挙げられる．本研究では前処理行列と BiCGSTAB 内部処理のデータ低精度化を行ったが，これは全て仮想的なものであり，また BiCGSTAB 内部処理では用いる全てのデータに対して適用している．ハードウェアやソフトウェアによる低精度データ型や低精度演算の実装は実行時間ベースの議論のために必要である．BiCGSTAB 内部処理において低精度化を適用する箇所の局所化や，前処理行列の低精度化との複合は，より効率的な低精度演算導入の足がかりとなる可能性があり，これらの精度を性能パラメタとした自動チューニングの適用も考えられる．また，SuiteSparse Matrix Collection などの実問題に近い疎行列を用いた数値実験が考えられる．今回用いたテスト行列はプログラムによって生成した行列であり，一定のパターンとシンプルな構造を有している．よって，より普遍的な議論を行うには様々な行列を用いる必要がある．

謝辞: 本研究の一部は JSPS 科研費 JP23K11126 の助成により実施した．

参考文献

[1] 深谷 猛，Zhao Yingqui，岩下 武史: ILU(0)前処理付き GMRES(*m*)法に対する低精度計算の導入可能性の検証，情報処理学会研究報告，Vol.2023-HPC-102 No.36, 2023.

[2] Yingqi Zhao, Takeshi Fukaya, Takeshi Iwashita, Numerical Behavior of Mixed Precision Iterative Refinement Using the BiCGSTAB method, Journal of Information Processing, Vol.31 860-874, 2023.