2007－HPC－111 (25)
2007／8／2

シュールコンプリメントを用いた大規模な近似逆行列の計算

張 臨 傑† 渡邉 慶太郎† 野 寺 隆††

大規模で非対称な係数行列を持つ連立1次方程式の近似解法には，クリロフ部分空間に基づく様々な反復解法が存在する．通常，このような反復法の収束を向上させるためには，方程式の前処理を利用することになる．近年，大規模な係数行列の前処理として，近似逆行列を用いる場合が増えている．本稿では部分構造法の観点に立ち，グラフ分割を利用して係数行列を再構成し，部分行列の計算に AISM 法 [*SIAM J. Sci. Comput.*, Vol. 25, No. 2, pp. 701–715 (2003)] による近似逆行列の応用を考える．さらに，シュールコンプリメント計算における速度向上を解析し，それを予測する手法を提案する．最後に，算法の実装を行い，数値実験により提案した手法の有効性を示す．

Computation of Large Approximate Inverse Matrix using Schur-Complement

LINJIE ZHANG ,† KEITARO WATANABE †
and TAKASHI NODERA††

There are lots of Krylov subspace iterative methods for the approximate solution of large sparse nonsymmetric linear systems of equations. To solve the large linear systems, we can usually apply an iterative method on the preconditioned equation to improve the convergence of iterative method. In recently, approximate inverse strategies are useful for the preconditioning of iterative method, for solving the large linear systems. In this paper, we make the computation strategy using Schur-complement from the view of substructure, using graph partitioning. We consider to apply AISM method [*SIAM J. Sci. Comput.*, Vol. 25, No. 2, pp. 701–715 (2003)] on the derived sub-matrices. We also analyze its Speed-Up in detail and give a way for predicting the Speed-Up. At last, we implement the proposed algorithm and also show numerical experiments for sufficiently large problem.

1. はじめに

通常，係数行列に持つ大規模な連立1次方程式:

$$A\boldsymbol{x} = \boldsymbol{b}, \quad A \in \boldsymbol{R}^{n\times n}, \quad \boldsymbol{x}, \boldsymbol{b} \in \boldsymbol{R}^n \qquad (1)$$

の近似解を求める反復解法は，クリロフ部分空間を構成する算法である．ただし，行列 $A \in R^{n\times n}$, $\boldsymbol{r}_0 \in R^n$ とすればクリロフ部分空間は，$m = 1, 2, \ldots, n$ に対して次のようになる．

$$\mathcal{K}_m(A, r_0) = \mathrm{span}\{r_0, Ar_0, \ldots, A^{(m-1)}r_0\}$$

クリロフ部分空間法には様々なものがあり[13]，GMRES(m) 法[14] はその中の1つの反復解法である．通常，式 (1) をクリロフ部分空間法で解く場合には，残差ノルムの収束を改善するために，式 (1) に対する行列の前処理が広く用いられている[13] [pp. 102-103]．行列の前処理とは，式 (1) に対する次式のような行列変換を行うことである．すなわち，行列 $M \in \boldsymbol{R}^{n\times n}$ を計算し，式 (1) の係数行列 A の右側から掛けて

$$AM\boldsymbol{y} = \boldsymbol{b}, \quad \boldsymbol{x} = M\boldsymbol{y} \qquad (2)$$

もしくは左側から掛けて

$$MA\boldsymbol{x} = M\boldsymbol{b}$$

のように係数行列の変換をして近似解を計算する手法である．また，式 (2) の右前処理は，疑似残差 $M(b - Ax)$ ではなく，正式な残差 $b - Ax$ を計算することになるので，一般的に広く使われている．従って，本稿でも右前処理を用いることにする．

近年，式 (1) に対して近似逆行列前処理行列 $M \approx A^{-1}$ を用いる手法が並列計算環境において良く用いられている．近似逆行列を求める算法は数多く提案されており[13] [pp. 997–1005]，不完全行列分解 (ILU 分解) と同様に広く用いられている[11]．森屋ら[12] ば，AISM 法[4] は多くの場合，前処理としての性能が安定していることを示し，従来並列計算に不向きといわれていた AISM 法の部分的な並列化を提案した．しかし，係数行列の次元が大きい場合，その近似逆行列を計算するために，膨大な計算時間が必要となるケースがしばしばある．本稿では，グラフ分割，或いは領域分割を利用して係数行列を再構成し，シュールコンプリメントを構成して計算を行う．さらに，シュールコ

† 慶應義塾大学大学院理工学研究科，Graduate School of Science and Technology, Keio University
†† 慶應義塾大学理工学部，Faculty of Science and Technology, Keio University

ンプリメントを用いた計算による速度向上を解析し，速度向上を予測する手法を与える．最後に計算機に実装を行い，この手法の有効性を示す．

2章は，グラフ分割を利用して係数行列を再構成し，シュールコンプリメントを利用した逆行列の計算法について述べる．3章は，Sherman-Morrison 公式に基づく AISM 法について述べる．4章は速度向上を予測する公式を提案する．5章は数値実験について述べ，提案した算法の有効性を示す．6章は総括と結論を述べる．

2. シュールコンプリメントを用いた近似逆行列の計算

シュールコンプリメントを用いた近似逆行列計算とは，グラフ分割や領域分割を利用して係数行列の要素を

$$P^TAP = \begin{bmatrix} A_1 & & & B_1 \\ & \ddots & & \vdots \\ & & A_p & B_p \\ C_1 & \cdots & C_p & A_s \end{bmatrix} \tag{3}$$

のように再構成し，その逆行列をシュールコンプリメントを用いて計算する手法である．ただし，P は置換行列である．本稿では，一般問題にも適用できるように，グラフ分割を利用する．

今，行列 $A \equiv (a_{ij}) \in \boldsymbol{R}^{n\times n}$ と対応する無向グラフを $G =< V, E >$ とする．ただし，$V = \{1,\ldots,n\}$ はグラフに属するノードの集合であり，E はエッジ $\{< i, j > \big| i, j \in V,\ a_{ij} \neq 0\}$ の集合である．グラフ分割によってグラフ G をほぼ均等に p 個に分割すると，得られる部分グラフ G_i に属するノードは内部点と境界点に分けられる．内部点とは，ほかの部分グラフ G_j $(j \neq i,\ j = 1,\ldots,p)$ に属する頂点との間に，エッジのないノードである．内部点以外の点は境界点である．ここで，内部点，境界点の順番で再オーダーリングを行い，行列 A にも新しい番号に従って行と列の交換を行うと，行列 A は式 (3) のような縁付きの 規則正しいブロック対角行列に変換できる[6]．また，A_i の次元は部分グラフ G_i の内部点の数と一致し，A_s の次元は境界点の総数と一致する．

シュールコンプリメントを用いた計算では，式 (3) の逆行列 $(P^TAP)^{-1}$ を次式のように分解する．

$$\begin{bmatrix} A_1^{-1} & & & E_1 \\ & \ddots & & \vdots \\ & & A_p^{-1} & E_p \\ & & & S^{-1} \end{bmatrix} \begin{bmatrix} I_1 & & & \\ & \ddots & & \\ & & I_p & \\ F_1 & \cdots & F_p & I_s \end{bmatrix} \tag{4}$$

ただし， $E_i = -A_i^{-1}B_iS^{-1}$, $F_i = -C_iA_i^{-1}$ であり，$S = A_s - \sum_{i=1}^{p} C_iA_i^{-1}B_i$ はシュールコンプリメントである．I_i と I_s はそれぞれ A_i，A_s と同次元の単位行列である．右側の A_i^{-1} と S^{-1} をそれぞれの逆行列 M_i と M_s で近似すると，P^TAP の近似逆行列 $(P^TAP)^{-1}$ は次式のようになる．

$$\begin{bmatrix} M_1 & & & \bar{E}_1 \\ & \ddots & & \vdots \\ & & M_p & \bar{E}_p \\ & & & M_s \end{bmatrix} \begin{bmatrix} I_1 & & & \\ & \ddots & & \\ & & I_p & \\ \bar{F}_1 & \cdots & \bar{F}_p & I_s \end{bmatrix}$$

ただし，$M_i \approx A_i^{-1}$，$M_s \approx \bar{S}^{-1}$，$\bar{E}_i = -M_iB_iM_s$，$\bar{F}_i = -C_iM_i$ である．ここで，$\bar{S} = A_s - \sum_{i=1}^{p} C_iM_iB_i$ は近似シュールコンプリメントである．

計算を行う際に，最初に A_i の逆行列 A_i^{-1} を次章で説明する AISM 法で計算する．そして，シュールコンプリメントの逆行列 S^{-1} を AISM 法で計算する．$\bar{E}_i$，$\bar{F}_i$ を明示的に計算しない．そのアルゴリズムを図 1 で示す．

```
1:  compute M_i ≈ A_i^{-1}, i = 1,...,p
2:  compute S̄_i = C_i M_i B_i, i = 1,...,p
3:  compute S̄ = A_s − Σ_{j=1}^{p} S̄_j
4:  compute M_s ≈ S̄^{-1}
```

図 1 近似逆行列の計算

3. AISM 法

Bru ら[4] によって提案された AISM 法は，次の Sherman-Morrison 公式を利用して，係数行列 A の近似逆行列を求める方法である．

[Sherman-Morrison 公式]

与えられた正則行列 $B \in \boldsymbol{R}^{n\times n}$ とベクトル $\boldsymbol{x} \in \boldsymbol{R}^n$, $\boldsymbol{y} \in \boldsymbol{R}^n$ に対して，もし $r = 1 + \boldsymbol{y}^TB^{-1}\boldsymbol{x} \neq 0$ なら，行列 $A = B + \boldsymbol{x}\boldsymbol{y}^T$ も正則行列であり，その逆行列は

$$A^{-1} = B^{-1} - r^{-1}B^{-1}\boldsymbol{x}\boldsymbol{y}^TB^{-1} \tag{5}$$

となる．◇

さらに，ベクトル $\{\boldsymbol{x}_k\}_{k=1}^{n} \in \boldsymbol{R}^n$，$\{\boldsymbol{y}_k\}_{k=1}^{n} \in \boldsymbol{R}^n$ と行列 $A_0 \in \boldsymbol{R}^{n\times n}$ は，次式を満たすものとする．

$$A = A_0 + \sum_{k=1}^{n} \boldsymbol{x}_k\boldsymbol{y}_k^T \tag{6}$$

ここで，式 (6) において $A_k = A_0 + \sum_{i=1}^{k} \boldsymbol{x}_i\boldsymbol{y}_i^T$ と置くと，$A_{k+1} = A_k + \boldsymbol{x}_{k+1}\boldsymbol{y}_{k+1}^T$，$A_n = A$ となる．もし $\boldsymbol{x}_k, \boldsymbol{y}_k$ と A_k が Sherman-Morrison 公式の条件を満たすならば，式 (5) を n 回適用すると，係数行列 A の逆行列は次式で計算できることになる．

$$A^{-1} = A_n^{-1} = A_0^{-1} - \sum_{k=1}^{n} r_k^{-1}A_{k-1}^{-1}\boldsymbol{x}_k\boldsymbol{y}_k^TA_{k-1}^{-1}$$

この式を整理すると

for $k = 1, \ldots, n$ do
  $\boldsymbol{x}_k = \boldsymbol{e}_k,\ \boldsymbol{y}_k = (\boldsymbol{a}^k - s\boldsymbol{e}_k)^T$,
  $\boldsymbol{u}_k = \boldsymbol{x}_k,\ \boldsymbol{v}_k = \boldsymbol{y}_k$
  for $i = 1, \ldots, k-1$
    $\boldsymbol{u}_k = \boldsymbol{u}_k - \frac{(\boldsymbol{v}_i)_k}{sr_i}\boldsymbol{u}_i$
    $\boldsymbol{v}_k = \boldsymbol{v}_k - \frac{\boldsymbol{y}_k^T\boldsymbol{u}_i}{sr_i}\boldsymbol{v}_i$
  end for
  for $i = 1, \ldots, n$
    if $|(\boldsymbol{u}_k)_i| <$ tolU dropoff $(\boldsymbol{u}_k)_i$
    if $|(\boldsymbol{v}_k)_i| <$ tolV dropoff $(\boldsymbol{v}_k)_i$
  end for
  $r_k = 1 + (\boldsymbol{v}_k)_k/s$
end for

図 2 AISM の算法

$$A_0^{-1} - A^{-1} = \sum_{k=1}^{n} r_k^{-1} A_{k-1}^{-1}\boldsymbol{x}_k\boldsymbol{y}_k^T A_{k-1}^{-1} \quad (7)$$

となる．ここで，式 (7) の右辺を行列形式に書き換えると，次式のようになる．

$$A_0^{-1} - A^{-1} = \Phi\Omega^{-1}\Psi^T \quad (8)$$

ただし，$\Phi = [A_0^{-1}\boldsymbol{x}_1, A_1^{-1}\boldsymbol{x}_2, \ldots, A_{n-1}^{-1}\boldsymbol{x}_n]$, $\Omega^{-1} = \mathrm{diag}[r_1^{-1}, r_2^{-1}, \ldots, r_n^{-1}]$, $\Psi = [\boldsymbol{y}_1^T A_0^{-1}, \boldsymbol{y}_2^T A_1^{-1}, \ldots, \boldsymbol{y}_n^T A_{n-1}^{-1}]$ である．行列 Φ と Ψ を計算するには，A_k^{-1} を計算し，保存する必要がある．しかし，新たなベクトル $\boldsymbol{u}_k$, $\boldsymbol{v}_k$ を

$$\boldsymbol{u}_k := \boldsymbol{x}_k - \sum_{i=1}^{k-1} \frac{\boldsymbol{v}_i^T A_0^{-1}\boldsymbol{x}_k}{r_i}\boldsymbol{u}_i,$$

$$\boldsymbol{v}_k := \boldsymbol{y}_k - \sum_{i=1}^{k-1} \frac{\boldsymbol{y}_k^T A_0^{-1}\boldsymbol{u}_i}{r_i}\boldsymbol{v}_i$$

というように定義すれば，次式が成立する．

$$A_{k-1}^{-1}\boldsymbol{x}_k = A_0^{-1}\boldsymbol{u}_k, \quad (9)$$

$$\boldsymbol{y}_k^T A_{k-1}^{-1} = \boldsymbol{v}_k^T A_0^{-1}, \quad (10)$$

$$r_k = 1 + \boldsymbol{y}_k^T A_0^{-1}\boldsymbol{u}_k = 1 + \boldsymbol{v}_k^T A_0^{-1}\boldsymbol{x}_k. \quad (11)$$

ここで，式 (9) と式 (10) を式 (8) に代入すると

$$A_0^{-1} - A^{-1} = A_0^{-1}U\Omega^{-1}V^T A_0^{-1} \quad (12)$$

となる．ただし，$U = [\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n]$, $V = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n]$ である．

さらに，Bru ら[4] は，A_0, $\boldsymbol{x}_k$, $\boldsymbol{y}_k$ について，次のような選択を提案している．

$$A_0 = sI_n, \quad (s > 0),$$

$$\boldsymbol{x}_k = \boldsymbol{e}_k, \quad \boldsymbol{y}_k = (\boldsymbol{a}^k - \boldsymbol{a}_0^k)^T.$$

ただし，$I_n \in \boldsymbol{R}^{n\times n}$ は単位行列であり，$\boldsymbol{e}_k \in \boldsymbol{R}^n$ は I_n の k 番目の列ベクトルである．$\boldsymbol{a}^k$ と $\boldsymbol{a}_0^k$ は行列 A と A_0 の k 番目の行ベクトルである．ここで，A_0, $\boldsymbol{x}_k$, $\boldsymbol{y}_k$ を式 (12) に代入すると，次式が得られることになる．

compute $M_i = s^{-1}I_i - s^{-2}U_i\Omega_i^{-1}V_i^T$
compute $\bar{S}_i = C_i M_i B_i$
gather $\bar{S}_j$, $(j = 1, \ldots, p\ j \neq i)$
compute $\bar{S} = A_s - \sum_{j=1}^{p} \bar{S}_j$
compute $M_s = s^{-1}I_s - s^{-2}U_s\Omega_s^{-1}V_s^T$

図 3 シュールコンプリメントを用いた計算

$$A^{-1} = s^{-1}I_n - s^{-2}U\Omega^{-1}V^T \quad (13)$$

また，$\boldsymbol{u}_k$, $\boldsymbol{v}_k$ と r_k は，次式のようになる．

$$\boldsymbol{u}_k = \boldsymbol{x}_k - \sum_{i=1}^{k-1} \frac{(v_i)_k}{sr_i}\boldsymbol{u}_i, \quad (14)$$

$$\boldsymbol{v}_k = \boldsymbol{y}_k - \sum_{i=1}^{k-1} \frac{\boldsymbol{y}_k^T\boldsymbol{u}_i}{sr_i}\boldsymbol{v}_i \quad (15)$$

ただし，

$$r_k = 1 + (\boldsymbol{v}_k)_k/s \quad (16)$$

このように A_0, $\boldsymbol{x}_k$, $\boldsymbol{y}_k$ を選べば，A_0^{-1} と $\boldsymbol{u}_k$, $\boldsymbol{v}_k$ が簡単に求められるだけでなく，行列 U の正則性も保証できることになる．さらに，$s \notin \sigma(A)$ なら，行列 V も正則である．ただし，$\sigma(A)$ は行列 A のスペクトル半径である．なお，これらの詳細に関しては，Bru ら[4] を参照して欲しい．また，U と V の疎な性質を維持するために，非ゼロ要素の切り捨ての処理を行う必要がある．従って，次式のようになる．

$$A^{-1} \approx s^{-1}I_n - s^{-2}U\Omega^{-1}V^T. \quad (17)$$

ここで，AISM 法の算法をまとめると，図 2 のようになる．

4. 速度向上の予測

一般的に，与えられる行列に対して，p の増大につれ，A_i の次元は小さくなり，A_s の次元は大きくなる傾向がある．従って，p の増大につれ，A_i の近似逆行列を計算する時間は短くなるものの，近似シールコンプリメント $\bar{S}$ の近似逆行列を計算する時間は長くなる．従って，ノードの数 p を大きくしても，シュールコンプリメントを用いた計算による近似逆行列の計算時間は必ず減少できるとは限らない．ここでは，シュールコンプリメントを用いた計算を行う際の速度向上を解析し，速度向上を予測する手法を与える．

まず，シュールコンプリメントを用いた計算を行う場合の速度向上 E を次式で定義する．

$$E = \frac{T_{origin}}{T} \quad (18)$$

ただし，T_{origin} はシュールコンプリメントを用いた計算を行わず，AISM 法で A の近似逆行列 $M \approx A^{-1}$ を計算する時間であり，T はシュールコンプリメントを用いた計算を行う場合の計算時間である．

計算コストで E を書き直すと，次式になる．

$$E = \frac{\mathrm{Ct}(M)}{D + \mathrm{Ct}(\bar{S}) + \mathrm{Ct}(M_s)}$$

ただし，Ct( ) はカッコ内の行列を計算するコストを意味する．また，$\mathrm{Ct}(M)$ はシュールコンプリメントを用いた計算を行わない場合の $M \approx A^{-1}$ の計算コストである．$D = \sum_{i=1}^{p}(\mathrm{Ct}(M_i) + \mathrm{Ct}(C_i M_i B_i))$ であり，$\mathrm{Ct}(\bar{S})$ は計算した $C_i M_i B_i$ を使って，近似シュールコンプリメント $\bar{S}$ を計算するコストである．本稿においては，計算コストは掛け算の回数だけを考える．

計算した $C_i M_i B_i$ を使って，近似シュールコンプリメント $\bar{S}$ を計算する際に，掛け算を行われないため，$\mathrm{Ct}(\bar{S})$ はゼロとなる．従って，E は次式で近似することができる．

$$E = \frac{\mathrm{Ct}(M)}{D + \mathrm{Ct}(M_s)} \tag{19}$$

AISM 法で行列 A の近似逆行列を計算するための計算コストについて考える．式 (16) で $\boldsymbol{u}_k$, $\boldsymbol{v}_k$ と r_k を計算するコストは各々 $(N(U)+2)(k-1)$, $(N(V)+N(A)+2)(k-1)$ と 1 である．ここで，N( ) はカッコ内の行列の行ごとの非ゼロ要素数の平均値を意味する．更に次元の小さい項を取り除くと，AISM 法の計算コストは約

$$\frac{1}{2}\,(N(A)+N(U)+N(V)+4)\dim(A)^2$$

である．この事実に踏まえて，M, M_i と M_s を計算するコストは各々以下のようになる．

$$\begin{aligned}
\mathrm{Ct}(M) &\approx \frac{1}{2}\,\big(N(A)+N(U) \\
&\qquad +N(V)+4\big)\dim(A)^2 \\
\mathrm{Ct}(M_i) &\approx \frac{1}{2}\,\big(N(A_i)+N(U_i) \\
&\qquad +N(V_i)+4\big)\dim(A_i)^2 \\
\mathrm{Ct}(M_s) &\approx \frac{1}{2}\,\big(N(S)+N(U_s) \\
&\qquad +N(V_s)+4\big)\dim(A_s)^2
\end{aligned} \tag{20}$$

$C_i M_i B_i$ の計算コストは実装によって異なる．本稿では，$M_i = s^{-1}I_i - s^{-2}U_i\Omega_i^{-1}V_i^T$ を明示的に計算せず，U_i, V_i^T と対角行列 Ω_i を計算する．また，Ω_i^{-1} を V_i^T に合併する．従って，

$$\begin{aligned}
\mathrm{Ct}(C_i M_i B_i) \approx\ & \mathrm{N}(B_i)\dim(A_i)^2 \\
& +2\mathrm{N}(C_i)\dim(A_s)^2 \\
& +\mathrm{N}(U_i)\dim(A_i)\dim(A_s).
\end{aligned} \tag{21}$$

式 (19)，式 (21) と式 (22) によって，計算を行う際の速度向上を計算できる．また，$N(U)$, $N(V)$, $N(U_i)$, $N(V_i)$, $N(U_s)$, $N(V_s)$, $N(S)$ に経験値を代入すれば，速度向上を予測することもできる．5 章の数値実験において，速度向上の予測値と実際の計測値との比較を行い，この手法を有効性を示す．

5. 数値実験

本章において，以下のような数値実験を行うものとする．

(1) シュールコンプリメントを用いた計算を行わず，AISM 法で近似逆行列を計算する．

(2) グラフ分割を行い，シュールコンプリメントを用いた計算を行う際の速度向上を予測する．

(3) シュールコンプリメントを用いた計算を行い，速度向上について検証し，予測値と比較する．

ここで，AISM 法のパラメータ s は，Bru ら[4] に基づき $s = 1.5\times\|A\|_\infty$ の値を使用する．また，行列 U, V の非ゼロ要素の drop off の閾値は，それぞれ $\mathrm{tolU} = TOL$, $\mathrm{tolV} = TOL$, かつ $TOL = 0.1$ とする．近似シュールコンプリメントの計算については，非ゼロ要素の切り捨て処理は行わない．計算した近似逆行列を GMRES(m) 法[14),15)] の右前処理行列として適用し，近似解を計算する．

以下のステップでグラフ分割を行う．

(ステップ 1): 係数行列のグラフを構成し，pmetis[10)] のフォーマットに従ってファイルに出力する．

(ステップ 2): pmetis を実行させ，グラフを p 個に分割する ($p = 2, 4, \ldots, 16$).

算法の実装は SGI 社の計算機を利用し，以下の条件にて数値実験を行うものとする．

- 計算機：MIPS R12000, CPU (300MHz)
- OS：IRIX 6.5.22
- コンパイラ: gcc version 2.8.1
- メモリ：8GB
- プログラム言語：C 言語
- 計算精度：倍精度

残差ノルムの収束評価に関しては，連立 1 次方程式の初期近似解をゼロベクトルとし，GMRES(m) 法の収束判定条件を

$$\|\boldsymbol{r}_k\|_2 \,/\, \|\boldsymbol{b}\|_2 \le 1.0\times10^{-12} \tag{22}$$

と設定する．ただし，最大反復回数 20000 以内で条件 (22) を満たせなければ反復を打ち止めにする．また，GMRES(m) 法のリスタート周期は 50 に設定し，反復回数は Krylov 部分空間の次元が 1 つ増加する度に 1 回と数える．計算時間と速度向上とも小数点後 2 桁まで表示する．

5.1 数値例

正方領域 $\Omega = [0,\,1]\times[0,\,1]$ における次のような偏微分方程式の境界値問題を考える．

$$\begin{aligned}
-u_{xx} - u_{yy} + D\{(y-\tfrac{1}{2})u_x \\
+(x-\tfrac{1}{3})(x-\tfrac{2}{3})u_y\} - 43\pi^2 u = G(x,\,y) \\
u(x,\,y)|_{\partial\Omega} = 1 + xy.
\end{aligned}$$

ただし，右辺 $G(x,\,y)$ は真の解が $u = 1 + xy$ になる

表 1　シュールコンプリメントを用いて計算を行わない場合の結果

Tpre	I	Tgmres	Ttotal
3851.81	7861	5707.99	9559.82

表 2　グラフ分割の結果

p	各部分グラフの内部点の数	境界点の総数
2	18239/18239	386
4	9033/9024/9014/9013	780
6	5986/5892/5986/5992/5819/5975	1214
8	4384/4457/4365/4465/4439/4380/4448/4360	1566
10	3486/3557/3505/3446/3513/3456/3562/3478/3411/3536	1914
12	2864/2960/2898/2941/2852/2910/2846/2908/2926/2948/2843/2904	2064
14	2431/2522/2473/2489/2470/2467/2406/2514/2412/2466/2444/2516/2415/2477	2362
16	2173/2110/2198/2157/2155/2110/2202/2159/2156/2182/2083/2095/2168/2199/2169/2088	2460

ように決定する．領域 Ω を格子点数 192×192 に区切り，5 点中心差分で離散化すると得られる連立 1 次方程式の次元は 36,864 となる．Dh の値には 2^{-7} の値を設定する．ただし，h はきざみ幅であり $h = 1/193$ である．

最初に，シュールコンプリメントを用いて計算を行わずに，AISM 法で近似逆行列を計算し，計算した近似逆行列を GMRES(m) 法に適用し，近似解を計算した．式 (22) を満たすまでに要した反復回数と計算時間の内訳を**表 1** に示す．ただし，Tpre は近似逆行列の計算時間を表している．I と Tgmres は GMRES(50) 法の収束条件 (22) を満たすまでに要した反復回数と計算時間を表すが，Tgmres には前処理時間である Tpre は含まれていない．T_{total} は総処理時間であり，$T_{\text{total}} = T\text{pre} + T\text{gmres}$ の関係がある．

次に，グラフ分割を行い，その結果を**表 2** に示す．表の第 2 列目から，pmetis はグラフをほぼ均等に分割していることが分かった．また，p の増加につれ A_i の次元が減少し，A_s の次元が増大する傾向が確認することができる．

ここで，4 章で与えた手法でシュールコンプリメントを用いた計算を行う際の速度向上を予測する．$N(U)$, $N(V)$，$N(U_i)$，$N(V_i)$，$N(U_s)$，$N(V_s)$ と $N(S)$ の値について，係数行列 A の行ごとの非ゼロ要素数 $N(A)$ を使用する．その結果を**表 3** に示す．E はシュールコンプリメントを用いた計算を行う場合の速度向上を意味する．

次に，シュールコンプリメントを用いた計算を行い，近似逆行列の計算時間と速度向上を検証する．その結果を表 4 に示す．T は近似逆行列の計算時間であり，E は式 (18) に従って，計算した速度向上である．ただし，T_{origin} はシュールコンプリメントを用いた計算を行わず，AISM 法で A の近似逆行列 $M \approx A^{-1}$ を計算する時間である．つまり，表 1 の Tpre である．表 4 から，シュールコンプリメントを用いた計算を採用することにより，約 13.70 倍の速度向上が得られたことが分かる．

表 3　速度向上の予測

p	予測した速度向上
2	2.01
4	3.97
6	5.72
8	7.20
10	8.35
12	9.39
14	10.03
16	10.73

表 4　シュールコンプリメントを用いて計算を行う場合の数値実験の結果

p	T	E
2	1686.56	2.25
4	818.30	4.64
6	551.11	6.89
8	426.68	8.90
10	360.54	10.53
12	315.92	12.02
14	295.39	12.86
16	277.31	13.70

表 6　GMRES(50) 法の数値実験の結果

p	Tpre	I	Tgmres	Ttotal
2	1686.23	9586	9649.20	11335.21
4	819.42	7633	7525.76	8344.34
6	550.18	8536	8412.58	8960.75
8	424.22	8061	7932.99	8353.81
10	358.99	7458	7467.11	7820.85
12	313.82	8296	8407.83	8715.56
14	282.21	7858	8112.67	8396.79
16	274.73	7467	7878.50	8144.68

シュールコンプリメントを用いた際の主な演算の計算時間を**表 5** に示す．各列の意味は以下のようになる．

p: 式 (4) の p と一致

T_{M_i}: 行列 A_i の近似逆行列 M_i を AISM 法を用いて計算する時間

$T_{C_iM_iB_i}$: 近似シュールコンプリメント $\bar{S}$ にローカル成分 $C_iM_iB_i$ の計算時間

T_{M_s}: 近似シュールコンプリメント $\bar{S}$ の近似逆行列 M_s の計算時間

最後に，AISM 法のシュールコンプリメントを用いて計算した近似逆行列を GMRES(m) 法に適用し，近似解を計算した．式 (22) を満たすまでに要した反復回数と計算時間を**表 6** に示す．p はグラフ分割数であり，Tpre，I，Tgmres と T_{total} の意味は表 1 と同じである．反復回数はすべて表 1 より小さい．$p = 10$ の反復回数が一番少なかった．また，グラフ分割数を増やすと，前処理時間が短縮できたが，総処理時間はあまり変わっていない．また，一回反復あたりの時間が増えている．

6. 結　　論

グラフ分割を利用して，近似逆行列を計算するためのシュールコンプリメントを用いた手法を提案した．

表 5 シュールコンプリメントを用いて計算を行う場合の数値実験の詳細

p	$T_{M_i, i=1,\ldots,p}$	$T_{C_i M_i B_i, i=1,\ldots,p}$	T_{M_s}
2	822.23/823.41	21.14/19.51	0.22
4	191.59/192.13/191.30/191.32	13.55/11.91/11.66/13.90	0.83
6	81.70/79.53/82.39/82.52/78.42/82.81	9.94/12.17/8.67/9.16/13.08/8.59	2.00
8	43.62/45.41/42.70/44.73/45.57/43.80/44.40/42.48	9.36/8.16/10.15/7.47/7.71/8.95/8.45/10.17	3.38
10	27.34/28.39/27.49/26.61/27.84/26.88/28.58/26.90 25.98/28.31	8.40/7.40/7.76/8.92/7.74/8.82/7.17/7.96 9.41/7.26	5.20
12	18.36/19.23/18.50/19.39/18.01/18.73/17.90/18.78 18.40/19.51/17.76/18.54	7.67/6.79/7.45/6.62/7.92/6.92/8.01/7.06 6.95/6.40/7.89/6.91	5.98
14	13.11/13.84/13.36/13.50/13.40/13.38/12.63/14.02 12.56/13.31/12.86/14.00/12.90/13.27	7.45/6.54/7.27/7.11/7.21/7.19/7.79/6.72 7.83/7.34/7.42/6.56/7.65/6.93	7.95
16	10.04/9.67/10.52/10.06/10.07/9.76/10.61/10.11 10.07/9.79/9.35/9.52/10.31/10.62/10.06/9.41	6.81/7.11/6.51/6.83/6.58/7.13/6.25/6.77 6.65/6.68/7.27/7.18/6.64/6.21/6.57/7.31	8.49

この計算手法は全ての近似逆行列を計算する方法に適用できる．さらに，提案したシュールコンプリメントを用いて計算する際の速度向上を予測する簡単な手法を提案した．数値実験の結果から，この予測値は実際の速度向上と同じような傾向を示した．従って，提案した計算手法の構成とその速度向上の推定法は，近似逆行列を求める有効な処理法の1つとなり得る．また，表5や表6の結果から，それぞれのグラフ分割をした際の各々の計算を並列化することによって，総処理時間の短縮が期待できると考えられる．これは今後の課題である．

参 考 文 献

1) Benzi, M., Meyer, C. D., and Tůma, M.: A Sparse Approximate Inverse Preconditioner for the Conjugate Gradient Method, *SIAM J. Sci. Comput.*, Vol. 17, No. 5, pp. 1135–1149 (1996).
2) Benzi, M., and Tůma, M.: A Sparse Approximate Inverse Preconditioner for Nonsymmetric Linear Systems, *SIAM J. Sci. Comput.*, Vol.19, No. 3, pp. 968–994 (1998).
3) Benzi, M., Marín, J., and Tůma, M.: A Two-level Parallel Preconditioner Based on Sparse Approximate Inverse, *Iterative Methods in Scienti c Computation IV, D.R. Kincaid, A. C. Elster, eds. IMACS Series in Computational and Applied Mathematics*, IMACS, New Brunswick, NJ, pp. 167–178 (1999).
4) Bru, R., Cerdán, J., Marín, J., and Mas, J.: Preconditioning Sparse Nonsymmetric Linear Systems with the Sherman-Morrison Formula, *SIAM J. Sci. Comput.*, Vol. 25, No. 2, pp. 701–715 (2003).
5) Bruaset, A. M.: A Survey of Preconditioned Iterative Methods, *Pitman Research Notes in Mathematics Series*, No. 328, Longman Scientific & Technical, U. K (1995).
6) Chow, E., and Saad, Y.: Approximate Inverse techniques for block-partitioned matrices, *SIAM J. Sci. Comput.*, Vol. 18, No. 6, pp. 1657–1675 (1997).
7) Chow, E., and Saad, Y.: Approximate Inverse Preconditioners via Sparse-sparse Iterations, *SIAM J. Sci. Comput.*, Vol. 19, No. 3, pp. 995–1023 (1998).
8) Grote, M., and Huckel, T.: Parallel Preconditioning with Sparse Approximate Inverses, *SIAM J.Sci.Comput.*, Vol.18, No.3, pp.838–853 (1997).
9) Huckel, T.: Efficient Computation of Sparse Approximate Inverses, *Numerical Linear Algebra Appl.*, Vol. 5, pp. 57–71 (1998).
10) Karypis, G., and Kumar, V.: A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs, *SIAM J. Sci. Comput.*, Vol. 20, No. 1, pp. 359–392 (1998).
11) 森屋健太郎，野寺隆：並列性を考慮した大規模な線形システムの前処理，応用数理，Vol. 12, No. 1, pp. 14–28 (2002).
12) 森屋 (健太郎)，張 (臨傑)，野寺 (隆)：Sherman-Morrison 法の並列化による近似逆行列の計算，情報処理学会論文誌，Vol. 48, No. 3, pp. 1462–1478 (2007).
13) Saad, Y.: Iterative Methods for Sparse Linear Systems, PSW Publishing Co., Boston (1966).
14) Saad, Y., and Schultz, M. H.: GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.*, Vol. 7, No. 3, pp. 856–869 (1986).
15) Shimoncini, Y.: On the Convergence of Restarted Krylov Subspace Method, *SIAM J. Matrix Anal. Appl.*, Vol. 22, No. 2, pp. 430–452 (2000).
16) Naik, V. K.: A Scalable Implementation of the NAS Benchmark BT on Distributed Memory Systems, *IBM Systems Journal.*, Vol. 34, No. 2, pp. 273–291 (1995).
17) University of Florida Sparse Matrix Collection. [Online]http://www.cise.ufl. edu/research/sparse/matrices.