

マルチグリッド法を QCD 計算に適用する具体的な方法

1 はじめに

QCD 計算にマルチグリッド法を適用した論文 [1] では, M. Brezina らによって 2004 年に提案された α SA 法 [2] を簡略化し, QCD 計算に対して適用する手法が提案された. また, その有効性が数値実験から示されている. 以下では, α SA 法の基本的アイデアを示し, 論文 [1] で提案されている具体的なアルゴリズムについて記す.

2 α SA 法の基本的アイデア

非エルミート線形方程式

$$Ax = b, \quad A \in \mathbb{C}^{n \times n}, \quad x, b \in \mathbb{C}^n$$

に対する AMG 法は, 一般に, 粗い格子の方程式 $\hat{A}\hat{x} = \hat{b}$ の係数行列 \hat{A} および制限行列 R を

$$\hat{A} = RAP, \quad R = P^H$$

とした 2 段グリッド法を再帰的に行うことにより計算される. $A_1x_1 = b_1$ に対する AMG 法の反復式を $x \leftarrow \text{AMG}_1(x, b_1)$ と置くと, $x_l \leftarrow \text{AMG}_l(x_l, b_l), l = 1, 2, \dots, N_L - 1$ のアルゴリズムは Algorithm 1 のように書ける. ここで, Pre/Post-Smoothing は方程式に反復法を数反復計算し近似解を更新する操作であり, 通常は Jacobi 法, Gauss-Seidel 法, SOR 法などの定常反復法が用いられる.

一般に, AMG 法では $Ae \approx 0$ を満たす成分を空間的低周波成分 (代数的に滑らかな成分) とし, 係数行列の各要素の関係性を基に補完行列 P_l を設定する. 補完行列 P_l の設定法としては,

- **C/F Splitting:** 粗い格子に含む変数の集合と含まない変数の集合を分離することで P_l を定義する; 図 1(a) 参照.
- **Aggregation:** 変数の集合を, 重ならない複数の集合 (アグリゲート) に分離し, その組み合わせによって P_l を定義する: 図 1(b) 参照.

の 2 種類に大きく分けることができる.

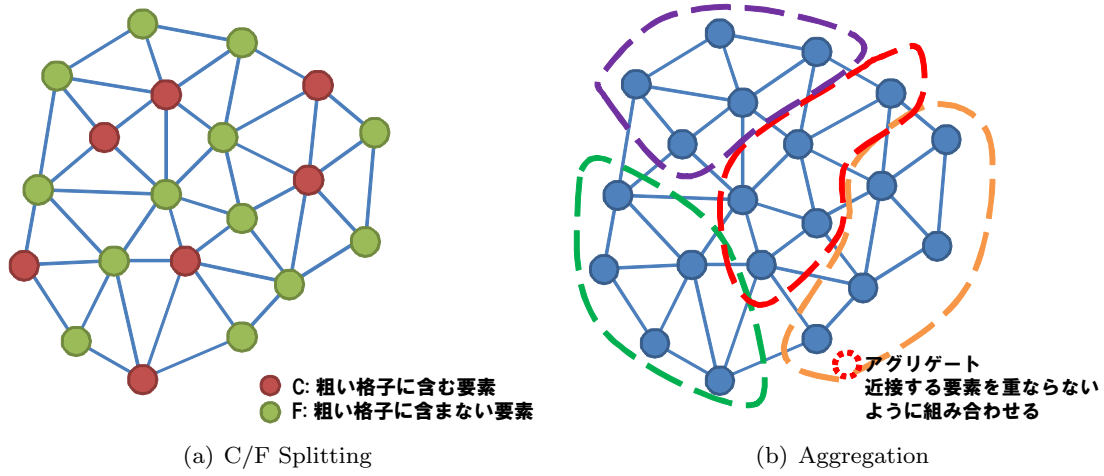


図1 C/F Splitting と Aggregation のイメージ

Algorithm 1 $\mathbf{x}_l \leftarrow \text{AMG}_l(\mathbf{x}_l, \mathbf{b}_l)$

- 1: **Pre-smoothing:** Apply ν iterations of pre-smoothing to $A_l \mathbf{x}_l = \mathbf{b}_l$
 - 2: **Coarse-grid correction:**
 - a: $\mathbf{b}_{l+1} = P_l^H (\mathbf{b}_l - A_l \mathbf{x}_l)$
 - b: **IF** $l + 1 < N_L$ **Then**
 - c: Set $\mathbf{x}_{l+1} = \mathbf{0}$
 - d: Solve (approximately) the coarse grid problem
 - e: $A_{l+1} \mathbf{x}_{l+1} = \mathbf{b}_{l+1}$, $A_{l+1} = P_l^H A_l P_l$
 - f: by $\mathbf{x}_{l+1} \leftarrow \text{AMG}_{l+1}(\mathbf{x}_{l+1}, \mathbf{b}_{l+1})$
 - g: **Else**
 - h: Solve the coarse grid problem directly
 - i: **End If**
 - j: Correct the solution on the level l : $\mathbf{x}_l \leftarrow \mathbf{x}_l + P_l \mathbf{x}_{l+1}$
 - 3: **Post-smoothing:** Apply μ iterations of post-smoothing to $A_l \mathbf{x}_l = \mathbf{b}_l$
-

α SA 法は、Aggregation 型の AMG 法の一つであり、同次の線形方程式 (homogenous system)

$$A\mathbf{x} = \mathbf{0} \tag{1}$$

をアグリゲーションに基づく AMG 法で繰り返し (近似的に) 解くことで、補完行列 P_l の改良を適応的に行う手法である。

3 QCD 計算に適用する具体的な方法

マルチグリッドの最大レベルを N_L , アグリゲートサイズを N_a , 最大ベクトル数を N_v , 反復回数を N_{iter} とする。この時、論文 [1] で提案された補完行列 P_l は、以下の 6 つの Step を $k = 1, 2, \dots, N_v$ と反復適用することで設定される。

Step 1: 同次の線形方程式 (1) を初期近似解 x_0 を乱数とし, 各レベルの補完行列を $P_{l,k-1}$ とした AMG 法を N_{iter} 反復適用し, 近似解 $v_k = x_{N_{\text{iter}}}$ を得る. ただし, $k = 1$ の時, AMG 法は pre/post-smoothing のみが行われる.

Step 2: $V_{1,k} = [V_{1,k-1}, v_k] \in \mathbb{C}^{n \times k}$ とする. ただし, $V_{1,1} = v_1$ であるとする.

Step 3: 変数の集合を, 変数の結びつきの強い要素数 N_a の集合 (アグリゲート) 毎に分離する. ここで, 各アグリゲート $\mathcal{A}_{1,k}^{(s)}$ は

$$\mathcal{A}_{1,k}^{(s)} \cap \mathcal{A}_{1,k}^{(t)} = \emptyset, \quad \text{および} \quad \bigcup_{s=1}^{n/N_a} \mathcal{A}_{1,k}^{(s)} = \{1, 2, \dots, n\} \quad (2)$$

を満たすとする. これに関して, 論文 [1] では,

In lattice QCD, the system is discretized on a uniform hypercubic lattice and the link matrices, U_x^μ , belong to $SU(3)$. This motivates the use of regular hypercubic coarsening, e.g., 4^4 blocking.

と記述されている.

Step 4: $V_{1,k}$ をサイズ N_a のアグリゲート (ブロック) 毎に分離し, 行列 $W_{1,k}$ を以下のように定義する:

$$W_{1,k} = [V_{1,k}^{(1)}, V_{1,k}^{(2)}, \dots, V_{1,k}^{(n/N_a)}] \in \mathbb{C}^{n \times n(k/N_a)}, \quad (3)$$

$$(V_{1,k}^{(s)})_{i,j} = \begin{cases} (V_{1,k})_{i,j} & i \in \mathcal{A}_{1,k}^{(s)} \\ 0 & i \notin \mathcal{A}_{1,k}^{(s)} \end{cases}, \quad s = 1, 2, \dots, n/N_a. \quad (4)$$

Step 5: $V_{1,k}^{(s)}$ を QR 分解:

$$V_{1,k}^{(s)} = Q_{1,k}^{(s)} R_{1,k}^{(s)}, \quad s = 1, 2, \dots, n/N_a \quad (5)$$

し, 列直交行列 $Q_{1,k}^{(s)} \in \mathbb{C}^{n \times k}$, $Q_{1,k}^{(s)H} Q_{1,k}^{(s)} = I$ と上三角行列 $R_{1,k}^{(s)} \in \mathbb{C}^{k \times k}$ を得る*1 また,

*1 長方形の QR 分解は Gram-Schmidt の直交化法や Householder 変換などにより計算可能であり, 例えば LAPACK では*GEQRF として実装されている. また, $V_{1,k}^{(s)}$ の QR 分解 (5) は, $V_{1,k}^{(s)}$ の $\mathcal{A}_{1,k}^{(s)}$ に属する行で構成される $N_a \times k$ 次の部分行列に対する QR 分解により効率的に計算できる.

$k = 1, 2, \dots$

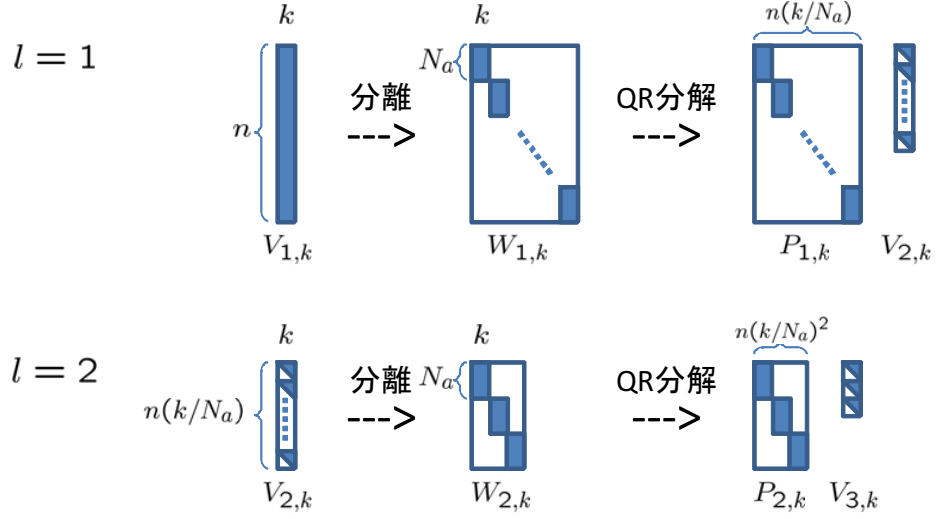


図 2 アグリゲート毎の分離および補完行列 $P_{1,k}$ の生成手順の概略. ただし, $\mathcal{A}_{1,k}^{(1)} = \{1, 2, \dots, N_a\}$, $\mathcal{A}_{1,k}^{(2)} = \{N_a + 1, N_a + 2, \dots, 2N_a\}, \dots$ とする.

$P_{1,k}, V_{2,k}$ をそれぞれ

$$P_{1,k} = [Q_{1,k}^{(1)}, Q_{1,k}^{(2)}, \dots, Q_{1,k}^{(n/N_a)}] \in \mathbb{C}^{n \times n(k/N_a)}, \quad V_{2,k} = \begin{bmatrix} R_{1,k}^{(1)} \\ R_{1,k}^{(2)} \\ \vdots \\ R_{1,k}^{(s)} \end{bmatrix} \in \mathbb{C}^{n(k/N_a) \times k} \quad (6)$$

と定義する. ここで,

$$V_{1,k} = P_{1,k} V_{2,k}, \quad P_{1,k}^H P_{1,k} = I$$

が満たされる.

Step 6: Step 3 から Step 5 を繰り返すことで, 各レベルの補完行列 $P_{l,k}, l = 1, 2, \dots, N_L - 1$ を生成する; 図 2 参照.

以上の手順は Algorithm 2 のようにまとめられる.

4 論文 [1] の数値実験で使用されたパラメータ

論文 [1] の数値実験では, 各種パラメータは表 1 に示されるように設定されている.

Algorithm 2 QCD 計算向けのマルチグリッド法の補完行列 P_l の生成

- 1: パラメータ設定
 - a: マルチグリッドのレベル: N_L , アグリゲートサイズ: N_b , 最大ベクトル数: N_v , 反復回数: N_{iter}
 - 2: メインの計算
 - a: **For** $k = 1, 2, \dots, N_v$ **Do**:
 - b: 初期ベクトル x_0 を乱数ベクトルに設定する.
 - c: N_{iter} 反復の AMG 法により, 近似解 v_k を得る: $v_k \leftarrow \text{AMG}_1^{N_{\text{iter}}}(x_0, \mathbf{0})$.
ここで, 各レベルの補完行列として $P_{l,k-1}$ を用いる.
 - d: $V_{1,k} = [V_{1,k-1}, v_k]$
 - e: **For** $l = 1, 2, \dots, N_L - 1$ **Do**:
 - f: 式 (2) を満たすように, アグリゲート $A_{l,k}^{(s)}$ を設定する.
 - g: 式 (3) から式 (6) に基づき, 補完行列 $P_{l,k}$ および行列 $V_{l+1,k}$ を計算する.
 - h: **End For**
 - i: **End For**
 - 3: 出力
 - a: $P_l \leftarrow P_{l,N_v}, l = 1, 2, \dots, N_L - 1$
-

表 1 各種パラメータの設定

パラメータ		設定
マルチグリッドのレベル	N_L	3
アグリゲートサイズ	N_a	4^4
最大ベクトル数	N_v	20
反復回数	N_{iter}	不明 (文献中で記述なし)
Pre-smoothing		なし
Post-smoothing		4 反復の GMRES 法 (減速パラメータ $\omega = 0.9$)
Krylov 部分空間法		GCR(8) 法

参考文献

- [1] R. Babich, J. Brannick, R. C. Brower, M. A. Clark, T. A. Manteuffel, S. F. McCormick, J. C. Osborn and C. Rebbi, Adaptive multigrid algorithm for the lattice Wilson-Dirac operator, Phys. Rev. Lett., 105 (2010), 201602.
- [2] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick and J. Ruge, Adaptive smoothed aggregation (α SA), SIAM J. Sci. Comput., 25 (2004), 1896–1920.