1B-7

並列GPGPUを利用した大規模行列に対する行列ランク最小化問題の解法

小西　克巳，坂本　亮（工学院大学）

1　はじめに

本稿では、大規模な行列ランク最小化問題に対するGPGPU を利用した並列化アルゴリズムを提案する。行列ランク最小化問題とは，ある制約下で変数となる行列のランクを最小にするような行列を求める問題である．この問題は，制御系同定におけるモデル次数の低次元化 [3]，データマイニングで利用される協調フィルタリング [?]，画像修復 [1] や動画像修復 [2] などの多くの問題へ応用されている．

行列ランク最小化問題は組合せ的な難しさを含み，一般には NP 困難な問題であるため，短い時間で厳密解を求めることは難しい．そこで，いくつかの発見的手法，例えば，行列の核ノルム (nuclear norm) 最小化に基づく方法 [3]，PowerFactorization アプローチ [4]，SVT(singular value thresholding) アルゴリズム [5]，定点反復法に基づくアルゴリズム [6] などが提案されている．しかしながら，強調フィルタリング等の実アプリケーションへの応用では，数億×数万の密な行列のランク最小化が望まれており，このような大規模な行列に対する手法は，計算資源の制約から提案されていない．そこで本稿では，すでに提案している NSAO (null space based alternating optimization) [7] に基づき，並列 GPU を利用した大規模行列のランク最小化問題の解法を提案する．

2　行列ランク最小化問題に対するNSAO法

本節では，すでに提案している行列ランク最小化問題に対する NSAO 法 [7] について説明する．ここでは，以下のような Matrix Completion と呼ばれる行列ランク最小化問題を扱う．

$$\begin{array}{ll}\text{Minimize} & \mathbf{rank}X \\ \text{subject to} & X_{ij} = M_{ij},\ \forall (i,j) \in \mathcal{I}\end{array} \tag{1}$$

ただし，$X \in \boldsymbol{R}^{m\times n}$ は設計変数であり，$m \le n$ とする．また，行列 M および添え字集合 $\mathcal{I}$ は与えられたものとする．この問題は一般には NP 困難な問題であり，解くのは難しい．行列 X のランクを最小化することは，その行列の零空間の次数を最大化することに等しいことから，問題 (1) は以下の問題と等価である．

$$\begin{array}{ll}\text{Maximize} & \mathbf{rank}W \\ \text{subject to} & XW = \mathbf{0}_{m,n},\ X_{i,j} = M_{i,j}\end{array} \tag{2}$$

ただし，$W \in \boldsymbol{R}^{n\times n}$ と $X \in \boldsymbol{R}^{m\times n}$ が設計変数であり，$\mathbf{0}_{m,n}$ は $m \times n$ の零行列を表す．この問題は問題 (1) と本質的には同じ難しさの問題であるが，目的関数のみに注目すると，ランクの小さい行列を求めることは組合せ的な困難があるのに対し，ランクの大きい行列を求めることは容易である．そこで NSAO 法では，問題 (2) に対する近似解法を与える．

Algorithm 1 NSAO-GPM

Input: $X \in \boldsymbol{R}^{m\times n}$, $\gamma > 0$, $\eta > 1$
repeat
$D_\Phi \leftarrow \gamma W + X^T X W;\ F_\Phi \leftarrow P_\Phi\left(W - 2D_\Phi\right) - W$
$\alpha_\Phi \leftarrow -\mathbf{Tr}\left(D_\Phi F_\Phi\right) / \|X^T F_\Phi\|_F^2$
$W \leftarrow W + \alpha_\Phi F_\Phi$
$D_\Omega \leftarrow XWW^T;\ F_\Omega \leftarrow P_\Omega\left(X - 2D_\Omega\right) - X$
$\alpha_\Omega \leftarrow -\mathbf{Tr}\left(D_\Omega^T F_\Omega\right) / \|F_\Omega W\|_F^2$
$X \leftarrow X + \alpha_\Omega F_\Omega$
$\gamma \leftarrow \gamma/\eta$
until termination criterion is satisfied
Output: low-rank solution X

問題 (2) に対し，以下の緩和問題を考える．

$$\begin{array}{ll}\text{Minimize} & \|W\|_F^2 \\ \text{subject to} & W_{ii} = 1,\ XW = \mathbf{0}_{m,n},\ X_{i,j} = M_{i,j}\end{array} \tag{3}$$

$W_{ii} = 1$ の制約下で $\|W\|_F^2$ を最小化した場合，自明な解として単位行列が得られ，W のランクは最大化される．ゆえに，上記問題は問題 (2) の良い近似解を与えることが期待される．問題 (3) は制約 $XW = \mathbf{0}_{m,n}$ を含むため難しい問題である．そこで，以下のようなラグランジュ緩和問題を考える．

$$\begin{array}{ll}\text{Minimize} & f_\gamma(W,X) \\ \text{subject to} & W_{ii} = 1,\ X_{i,j} = M_{i,j},\end{array} \tag{4}$$

ただし，関数 $f_\gamma(W,X)$ は，

$$f_\gamma(W,X) = \gamma\|W\|_F^2 + \|XW\|_F^2$$

と定義され，$\gamma > 0$ である．上記問題は目的関数が凸関数でないため厳密に解くことは難しいが，X または W を定数とすると凸最適化問題として解けるため，X と W を交互に固定化することにより精度の高い近似解を得ることができる．射影勾配法 (gradient projection method, GPM) を適用することにより，Algorithm 1 に示す NSAO-GPM 法を得る．ただし，P_Φ は行列の対角成分を全て 0 に置き換え，P_Ω は行列の成分のうち集合 $\mathcal{I}$ に属する成分のみを 0 に置き換える射影である．

3　提案手法

NSAO-GPM 法は全ての計算が行列の和と積のみであるという特徴を持つ．文献 [7] では，行列サイズが大き

Algorithm 2 paralel NSAO

Input: $X \in \boldsymbol{R}^{k_p p \times k_q q}$, i_{max}
  $X_{sum} \leftarrow \mathbf{0}$
  for $i = 1$ to i_{max} **do**
    permute rows and columns of X randomly
    divide X into $k_p k_q$ matrices X^i of size $p \times q$
    parallel apply Algorithm 1 to X^i
    reconstruct X from X^i
    $X_{sum} \leftarrow X_{sum} + X$
  end for
  $X \leftarrow X_{sum}/i_{max}$
Output: low-rank solution X

い場合を考え，それぞれの行列積を複数 GPU によって並列化し，計算を高速化している．しかしながら，CPU と GPU 間の通信コストが大きいため，行列サイズが大きいほど高速化されなくなる．そこで本稿では，より小さな問題に問題を分割し，複数の GPU で可能な限り独立して並列計算可能なアルゴリズムを提案する．

本稿では Algorithm 2 を提案する．ただし，$m = k_p p$, $n = k_q q$ と仮定し，行列全体を $k_p k_q$ 個に分割できると仮定している．行列の行と列の順序を入れ替えても行列ランクは変化しないことに注目し，提案アルゴリズムでは行と列をランダムに入れ替え行列分割を行い，分割された各行列に対して Algorithm 1 を並列に適用している．計算精度を上げるため，複数回の試行を繰り返し平均を計算している．

4 数値実験

提案する並列アルゴリズムの有効性を確認するため，数値実験を行った．全ての実験において，Algorithm 1 では，$\eta = 1.1$, $\gamma = 1 \times 10^{-2}$ を用いた．復元するランク r の行列 X は $n \times n$ の正方行列とし，正規分布に従い生成した行列 $Y_1 \boldsymbol{R}^{m \times r}$ と行列 $Y_2 \boldsymbol{R}^{r \times n}$ を用いて $X = Y_1 Y_2$ を計算して生成した．ただし，$\|X\| = 1$ となるように正規化した．既知行列 M は，X の成分のうち 20% が既知であるとし，一様乱数を用いて生成した．なお全ての計算は，CPU：Core i7 3.4GHz，RAM：16GB，および，NVIDIA 製 Tesla を搭載した PC において MATLAB および MATLAB 用 GPU 計算ライブラリである Jacket を利用した．

まず最初に，CPU，1 枚の GPU，2 枚の GPU での計算時間の比較実験を行った．GPU 1 枚の場合は行列分割せずに Algorithm 1 を適用した．GPU 2 枚の場合は，行列を $4 \times 4 = 16$ 分割して計算し，$i_{max} = 1$ とした．実験結果を表 1 に示す．GPU 1 枚の場合はメモリ制約により，$n = 4000$ までしか計算できない．

次に i_{max} の値と計算精度の関係を調べるため，$n = 2000$ の行列に対し，行列を $2 \times 2 = 4$ 分割して GPU 2 枚で並列計算を行った．結果を図 1 に示す．誤差は，最適解 X_{opt} と得られた解 X^* に対して error $= \|X_{opt} - X^*\|_F / \|X_{opt}\|_F$ と定義される．繰り返し数を増やすことで，計算精度が向上することが確認できる．

Table 1: 実験 1．計算時間 [sec].

n	r	CPU	GPU x1	GPU x2
1000	10	35.42	5.180	5.758
2000	10	268.1	26.98	13.42
4000	10	2013	162.23	44.80
8000	10	16042	-	509.25
10000	10	30959	-	2062.2

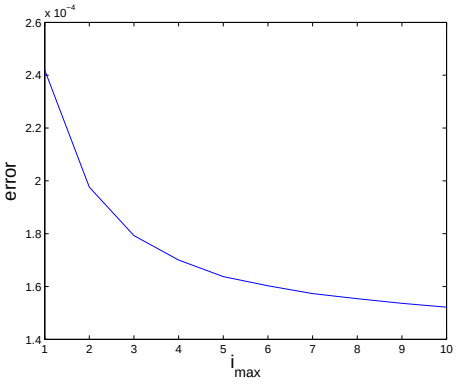

Fig. 1: 実験 2．計算精度.

5 まとめ

本稿では大規模な行列ランク最小化問題を扱い，すでに提案している NSAO 法の並列アルゴリズムを提案した．元の問題を分割し，より小さい問題として扱う手法である．数値実験により，提案手法の有効性を示した．

参考文献

1) T. Takahashi, K. Konishi and T. Furukawa, Reweighted l2 norm Minimization Approach to Image Inpainting Based on Rank Minimization, Proc. of IEEE MWSCAS2011 (2011)
2) T. Ding, M. Sznaier, and O.I. Camps, A Rank Minimization Approach to Video Inpainting, Proc. of IEEE International Conference on Computer Vision, 1/8 (2007)
3) K. Mohan and M. Fazel, Iterative reweighted least squares for matrix rank minimization, Proc. of the Allerton Conference on Communications, Control, and Computing, 653/661 (2010)
4) J. P. Haldar and D. Hernando, Rank-Constrained Solutions to Linear Matrix Equations Using PowerFactorization, IEEE Signal Procesisng Letters, vol. 16, no. 7, 584/587 (2009)
5) J.-F. Cai, E. J. Candès and Z. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optimiz., vol. 20, no. 4, 1956/1982 (2010)
6) S. Ma, D. Goldfarb and L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization, Mathematical Programming, vol. 128, no. 1-2, 321/353 (2011)
7) K. Konishi, Parallel GPU Implementation of Null Space based Alterating Optimization Algorithm for Large-Scale Matrix Rank Minimization, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2012 (to appear)