A MULTIGRID METHOD FOR
THE MAXIMAL CORRELATION PROBLEM

XIN-GUO LIU AND KUN WANG
School of Mathematical Science
Ocean University of China, Qiaodao 266100, China

(Communicated by Zhongzhi Bai)

ABSTRACT. The maximal correlation problem (MCP) aiming at optimizing correlations between sets of variables plays an important role in several areas of statistical applications. Several algorithms have been proposed for MCP. However, these algorithms are not guaranteed to find a global optimal solution of MCP. Employing an idea of the multigrid method for PDEs, a new method for MCP is proposed in the present paper. Numerical tests are carried out and suggest superior performance of the new method to the others in finding a global optimal solution of MCP.

1. Introduction. The maximal correlation problem (MCP) as a generalization of the canonical correlation analysis [9] aims at assessing the relationship between sets of random variables. Suppose $A \in \mathbb{R}^{n \times n}$ is a symmetric and positive definite matrix, and $P = \{n_1, n_2, \ldots, n_m\}$ is a set of positive integers with $\sum_{i=1}^{m} n_i = n$. Let $\Sigma_m = \{x = (x_1^\top, \ldots, x_m^\top)^\top \in \mathbb{R}^n : x_i \in \mathbb{R}^{n_i}, \|x_i\|_2 = 1, \text{ for } i = 1, 2, \ldots, m\}$ and $\rho(x) = x^\top Ax$. The MCP is formulated as the following optimization problem:

$$\begin{align*}
\text{maximize } & \rho(x), \\
\text{subject to } & x \in \Sigma_m.
\end{align*}$$

Upon employing the Lagrange multiplier theory, it is easy to see that the first order necessary optimality condition for (1.1) is the existence of real scalars $\lambda_1, \ldots, \lambda_m$ and a vector $x \in \Sigma_m$ that solve the system of equations:

$$\begin{align*}
Ax &= \Lambda x, \\
x &\in \Sigma_m,
\end{align*}$$

where $\Lambda = \text{diag}\{\lambda_1 I_{n_1}, \ldots, \lambda_m I_{n_m}\}$. This gives the multivariate eigenvalue problem (MEP), and $(\Lambda, x)$ satisfying (1.2) is called a solution to MEP.

A brief but comprehensive discussion on the statistical background of MCP can be found in Sun [11] and Chu and Watterson [1]. More applications of MEP can be found in [10] [13]. Generalizations of MCP can be found in [12]. An important result proved in [1] states that whenever the matrix $A$ has $n$ distinct eigenvalues, then there are precisely $\prod_{i=1}^{m} (2n_i)$ solutions to the MEP (1.2).

2000 Mathematics Subject Classification. Primary: 62H20; Secondary: 65F10.

Key words and phrases. Multivariate statistics, correlation, maximal correlation problem, multivariate eigenvalue problem, Horst method, Gauss-Seidel method, alternating projection method, global solution, starting point.

This research was supported in part by NSF of China (10971204, 11071228) and NSF of Shandong province (Y2008A07).
Obviously, a special case of MEP where \( m = 1 \) is the classical symmetric eigenvalue problem. The Horst method \([7]\) is a generalization of the classical power iteration. Its convergence is established later by Hu \([3] \), and independently by Chu and Watterson \([1]\). A Gauss-Seidel-type iteration is proposed in \([1]\) and analyzed later in \([17] \). As another generalization of the Horst method, the P-SOR iteration is proposed and analyzed by Sun \([11] \). These algorithms for the general MCP stop at solutions of the associated MEP for a related matrix \( A \). Recently, Grzegórski \([4] \), and independently Zhang and Liao \([18] \), proposes an alternating projection method (APM) (in \([18] \), called alternating variable method (AVM)). They prove that (i) the algorithm converges globally and monotonically to a solution of MEP under some assumptions, and (ii) any accumulation point satisfies a globally optimal condition of MCP. However, simple examples show that the APM also can not guarantee to converge to a global optimal solution of MCP (see Section 3 below).

In searching the global optimal solution of MCP, several achievements have been made by Hanafi and Ten Berge \([3] \), Zhang and Chu \([17] \), and Zhang, Liao and Sun \([19] \). For the general case, however, solving \((1.1)\) globally still remains very challenging. One idea suggested by Xu \([14] \) and independently by Zhang and Chu is to set a starting point strategy for the existing algorithms. Even though numerical experiment has demonstrated the substantial improvement in boosting the performance of the new algorithm to the others in finding a global optimal solution of \((1.1)\), we are still not satisfied with the current algorithms.

Employing an idea of the multigrid method for PDEs, in this paper, we propose a multigrid-type method for MCP. The rest of the paper is organized as follows. In the next section, we prove several properties of MCP. In Section 3 simple examples are presented to demonstrate that no algorithm so far can guarantee convergence to a global optimal solution of MCP. We propose a multigrid algorithm in Section 4. Finally, in Section 5 numerical tests are carried out and suggest superior performance of the new algorithm to the others in finding a global optimal solution of MCP.

2. Properties of MCP. Let \( A \in \mathbb{R}^{n \times n} \) be symmetric and positive definite. If \( P = \{n_1, n_2, \ldots, n_m\} \) is a set of positive integers with \( \sum_{i=1}^{m} n_i = n \), then we call \( P \) a partition of \( n \). Let \( P = \{n_1, n_2, \ldots, n_m\} \) be a partition of \( n \), and let \( P' = \{n_1, \ldots, n_{k-1}, \widehat{n}_k, n_{k+2}, \ldots, n_m\} \) with \( \widehat{n}_k = n_k + n_{k+1} \). Then we call \( P \) a finer partition of \( P' \).

Obviously, given a partition \( P = \{n_1, n_2, \ldots, n_m\} \) of \( n \), there are a series of partitions \( P_1, \ldots, P_m \), such that

(i) \( P_1 = \{n\} \), \( P_m = P \),

(ii) \( P_i \) is finer than \( P_{i-1} \) for \( i = 2, \ldots, m \).

Let \( P_i = \{t_1, \ldots, t_i\} \). Accordingly, let

\[ \Sigma_i = \{x = (x_1^\top, \ldots, x_i^\top)^\top \in \mathbb{R}^n : x_j \in \mathbb{R}^{t_j}, ||x_j||_2 = 1, \text{ for } j = 1, 2, \ldots, i\} \]

\[ \Omega_i = \{x = (x_1^\top, \ldots, x_i^\top)^\top \in \mathbb{R}^n : x_j \in \mathbb{R}^{t_j}, ||x_j||_2 \leq 1, \text{ for } j = 1, 2, \ldots, i\} \]

Theorem 2.1.

\[
\frac{1}{2} \max_{x \in \Sigma_i} \rho(x) \leq \max_{x \in \Sigma_{i-1}} \rho(x) \leq \max_{x \in \Sigma_i} \rho(x). \tag{2.1}
\]

Proof. It is proved by Sun \([11] \) that

\[
\max_{x \in \Sigma_i} \rho(x) = \max_{x \in \Omega_i} \rho(x).
\]
This implies that the second inequality in (2.1) holds. On the other hand, we note that 
$$\frac{1}{\sqrt{2}}x \in \Omega_{i-1}$$ for any \( x \in \Omega_i \). Then
$$\max_{x \in \Omega_i} \rho(x) = \max_{x \in \Omega_i} \rho(\frac{1}{\sqrt{2}}x) \leq \max_{y \in \Omega_{i-1}} \rho(y) = 2 \max_{x \in \Omega_{i-1}} \rho(x).$$
This completes the proof.

The following corollary presents bounds of the function value for MCP.

**Corollary 2.2.** Let \( A = (a_{ij})_{n \times n} \). Then
$$\lambda_{\max}(A) \leq \max_{x \in \Sigma_m} \rho(x) \leq \min\{m \lambda_{\max}(A), \mu\}, \quad (2.2)$$
where \( \lambda_{\max}(A) \) stands for the largest eigenvalue of \( A \), and
$$\mu = \max_{|d_1|=\cdots=|d_n|=1} \sum_{i,j=1}^n d_id_ia_{ij}.$$  

**Proof.** The first inequality in fact can be reformulated as
$$\max_{x \in \Sigma_1} \rho(x) \leq \max_{x \in \Sigma_m} \rho(x),$$
and directly obtained from Theorem 2.1. The inequality
$$\max_{x \in \Sigma_m} \rho(x) \leq \mu$$
is in fact equivalent to the following one:
$$\max_{x \in \Sigma_m} \rho(x) \leq \max_{x \in \Sigma_n} \rho(x),$$
and directly obtained from Theorem 2.1. Finally, by the Courant-Fisher minmax theorem (see, e.g., [3, Theorem 8.1.2]) one gets that
$$\rho(x) \leq m \lambda_{\max}(A).$$
This completes the proof.

The next theorem presents other bounds for the MCP value.

**Theorem 2.3.** Let \( P = \{n_1, n_2, \ldots , n_m\} \) be a partition of \( n \). Partition \( A \) into block forms according to \( P \) as follows,
$$A = \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mm} \end{pmatrix} \in \mathbb{R}^{n \times n},$$
with \( A_{ii} \in \mathbb{R}^{n_i \times n_i} \). Then
$$\sum_{i=1}^m \lambda_{\max}(A_{ii}) \leq \max_{x \in \Sigma_m} \rho(x) \leq \sqrt{ \sum_{i=1}^m \lambda_{\max}(A_{ii})^2}. \quad (2.3)$$

**Proof.** The first inequality is a consequence of [17, Theorem 3.2]. In order to prove the second inequality in (2.3), consider the Cholesky factorization of \( A \),
$$A = L^T L,$$
where \( L = [L_1, \cdots, L_m] \) with \( L_i \in \mathbb{R}^{n_i \times n_i} \). Then, for any \( x = (x_1^\top, \cdots, x_m^\top)^\top \in \Sigma_m \), one gets that
\[
\rho(x) = \|Lx\|_2^2 = \|L_1x_1 + \cdots + L_mx_m\|_2^2 \leq \left( \sum_{i=1}^{m} \|L_i\|_2 \right)^2.
\]
This inequality and the fact
\[
A_{ii} = L_i^\top L_i
\]
implicate that the second inequality in (2.3) holds. This completes the proof.

The following theorem presents a comparison of the lower bounds in (2.2) and (2.3).

**Theorem 2.4.**
\[
\lambda_{\max}(A) \leq \sum_{i=1}^{m} \lambda_{\max}(A_{ii}).
\]

**Proof.** Let
\[
Ax = \lambda_{\max}(A)x, \quad x \in \mathbb{R}^n, \quad \|x\|_2 = 1.
\]
Partition \( x \) according to \( \mathcal{P} \) into block form
\[
x = (x_1^\top, \cdots, x_m^\top)^\top, \quad x_i \in \mathbb{R}^{n_i}.
\]
Write \( x_i \) as
\[
x_i = t_i v_i, \quad v_i \in \mathbb{R}^{n_i}, \quad \|v_i\|_2 = 1, \quad t_i \in \mathbb{R}.
\]
Let
\[
V = \text{diag}(v_1, \cdots, v_m), \quad t = (t_1, \cdots, t_m)^\top.
\]
Obviously, one has that
\[
V^\top V = I_m, \quad \|t\|_2 = 1, \quad x = Vt.
\]
As a consequence, we see that
\[
(V^\top AV)t = \lambda_{\max}(A)t.
\]
And therefore,
\[
\lambda_{\max}(A) \leq \text{tr}(V^\top AV) = \sum_{i=1}^{m} v_i^\top A_{ii} v_i \leq \sum_{i=1}^{m} \lambda_{\max}(A_{ii}).
\]
This completes the proof.

The next corollary presents a comparison of the upper bounds in (2.2) and (2.3) under some assumptions.

**Corollary 2.5.** Suppose \( A_{ii} = I_{n_i} \) for \( i = 1, 2, \cdots, m \). Then
\[
\lambda_{\max}(A) \leq m.
\]

Corollary 2.5 demonstrates that under the assumptions \( A_{ii} = I_{n_i} \) for \( j = 1, \cdots, m \), the upper bound in (2.2) is sharper than the one in (2.3). On the other hand, it is worth pointing out that for general \( A \), \( m\lambda_{\max}(A) \) may be larger than the upper bound in (2.3) as the following example demonstrates.

**Example 2.1.** Let \( A = \begin{pmatrix} 1 & 1 \\ 1 & a \end{pmatrix}, \quad n_1 = n_2 = 1. \) For \( 0 < a < 1 \) or \( a > 5 \), it is easy to see that
\[
2\lambda_{\max}(A) > (\lambda_{\max}(A_{11}) + \lambda_{\max}(A_{22}))^2.
\]
The next example presents further information on the bounds in (2.2) and (2.3).

Example 2.2. Let \( A = \begin{pmatrix} 1 & a & \cdots & a \\ a & 1 & \cdots & a \\ \vdots & \vdots & \ddots & \vdots \\ a & a & \cdots & 1 \end{pmatrix} \), \( n = 2s \). If we choose \( m = s \) and \( n_i = 2 \) for \( i = 1, 2, \ldots, m, 0 < a < 1 \), then a computation gives that \( \lambda_{\max}(A) = (1-a) + na \), \( \lambda_{\max}(A_{ii}) = 1 + a \). Consequently, we have

\[
\frac{\lambda_{\max}(A)}{\sum_{i=1}^{m} \lambda_{\max}(A_{ii})} = \frac{2a + \frac{1}{m}(1-a)}{1 + a} \geq \frac{2a}{1 + a},
\]

and

\[
\frac{(\sum_{i=1}^{m} \lambda_{\max}(A_{ii}))^2}{m \lambda_{\max}(A)} = \frac{1 + a}{2a + \frac{1}{m}(1-a)}.
\]

One sees that when \( a \to 1 \) the bounds in (2.2) are close to the ones in (2.3). Furthermore, if we choose \( m = n \), then we see that \( \lambda_{\max}(A) \to m \) as \( a \to 1 \) and hence the bound in Corollary 2.3 is approximately sharp.

3. Global solution and the existing algorithms. As mentioned in Section II several effective algorithms for solving MEP have been proposed in literature. We now briefly describe these algorithms as follows. Consider the following splitting of \( A \):

\[
A = D + L + U,
\]

where \( D = \text{diag}(A_{11}, \ldots, A_{mm}), \) \( L = U^T \) is a strictly lower triangular matrix.

Horst algorithm:

\[
\Lambda^{(k)} x^{(k+1)} = Ax^{(k)}, \tag{3.1}
\]

where \( \Lambda^{(k)} = \text{diag}(\lambda_1^{(k)} I_{n_1}, \ldots, \lambda_m^{(k)} I_{n_m}) \). For details see Horst [7], Hu [8], and Chu and Watterson [1]. This algorithm defines a map on \( \Sigma_{m} \), denoted by \( \Phi_H \), so that \( x^{(k+1)} = \Phi_H(x^{(k)}) \).

Gauss-Seidel algorithm:

\[
(\Lambda^{(k)} - L) x^{(k+1)} = (D + U) x^{(k)}, \tag{3.2}
\]

where \( \Lambda^{(k)} = \text{diag}(\lambda_1^{(k)} I_{n_1}, \ldots, \lambda_m^{(k)} I_{n_m}) \). See Chu and Watterson [11], and Zhang and Chu [17]. This algorithm can be written as \( x^{(k+1)} = \Phi_{GS}(x^{(k)}) \).

P-SOR algorithm:

\[
(\Lambda^{(k)} M^{(k)} - \Omega^{(k)} L) x^{(k+1)} = [(I_n - \Omega^{(k)}) \Lambda^{(k)} + \Omega^{(k)} (D + U)] x^{(k)}, \tag{3.3}
\]

where \( M^{(k)} \) and \( \Omega^{(k)} \) are diagonal matrices. See Sun [11] for details. This algorithm can be expressed as \( x^{(k+1)} = \Phi_{PSOR}(x^{(k)}) \).

APM(or AVM [18]) algorithm:

\[
(\Lambda^{(k)} - D) x^{(k+1)} = L x^{(k+1)} + U x^{(k)} \tag{3.4}
\]

This algorithm can be expressed as \( x^{(k+1)} = \Phi_{APM}(x^{(k)}) \).

It is not difficult to verify the following results.

**Proposition 3.1.** Let \( (\Lambda^*, x^*) \) be a solution of the MEP (1.2) with \( \Lambda^* = \text{diag}(\lambda_1^* I_{n_1}, \ldots, \lambda_m^* I_{n_m}) \) satisfying \( \lambda_i^* \geq 0 \) for \( i = 1, 2, \ldots, m \). Then \( x^* \) is a fixed point of the maps \( \Phi_H, \Phi_{GS} \), and \( \Phi_{PSOR} \). Furthermore, if \( \lambda_i^* \geq \lambda_{\max}(A_{ii}) \) for \( i = 1, 2, \ldots, m \), then \( x^* \) is a fixed point of the map \( \Phi_{APM} \).
One consequence of Proposition 3.1 is that if \((\Lambda^*, x^*)\) is a solution of the MEP (1.2) but \(x^*\) is not a local maximizer of the MCP (1.1) and \(x^{(0)} = x^*\) is taken as starting point, then the sequence \(\{x^{(k)}\}\) generated by any of the algorithms mentioned above do not converge to a global maximizer of the MCP (1.1). Therefore, these algorithms do not guarantee to obtain a maximizer of MCP.

In fact, Chu and Wattrison [1] have observed that the computed solution by the Horst algorithm depends heavily on selection of the starting point. Similar observation has been made by Xu [14] for the P-SOR algorithm. The following examples demonstrate that the computed results of APM also depends on the starting point choice.

**Example 3.1.** Let \(A = \begin{pmatrix} a + 1 & 0 & 0 & 1 \\ 0 & a & 1 & 0 \\ 0 & 1 & b + 1 & 0 \\ 1 & 0 & 0 & b + 1 \end{pmatrix}, \ a > 0, \ b > 0, \ n_1 = n_2 = 2.\)

If take the \(x^{(0)} = (1, 0, 1, 0)^\top\) as starting point, then a computation demonstrates that the sequence \(\{\|Ax^{(k)} - \Lambda^{(k)}x^{(k)}\|_2\}\) does not converge to zero, where \((\Lambda^{(k)}, x^{(k)})\) are computed by the APM.

**Example 3.2.** Let \(A = \begin{pmatrix} a + 1 & 0 & 0 & 1 \\ 0 & a & 1 & 0 \\ 0 & 1 & b & 0 \\ 1 & 0 & 0 & b + 1 \end{pmatrix}, \ a > 0, \ b > 0, \ m = 2, \ n_1 = n_2 = 2.\)

Take \(x^{(0)}\) as Example 3.1, the sequence \((\Lambda^{(k)}, x^{(k)})\) generated by the APM converges to a solution \((\Lambda^*, x^*)\) of MEP, where \(\lambda_1^* = a + 1, \lambda_2^* = b + 1, \ x^* = (0, 1, 1, 0)^\top.\)

By [6], it is easy to see that \(x^*\) is not a global optimal solution of MCP.

**4. A multigrid method for MCP.** Taking advantage of the bounds established in Theorem 2.1 and employing an idea from the multigrid methods for PDEs [5], we propose in this section a multigrid-type method for MCP.

Let \(P^{(1)}, \ldots, P^{(L)}\) be partitions of \(n\) satisfying:

(i) \(P^{(1)} = \{n\}, \ P^{(L)} = \{n_1, \ldots, n_m\},\)

(ii) \(P^{(i)}\) as a partition of \(n\) is finer than \(P^{(i-1)}\) for \(i = 2, 3, \ldots, L.\)

Let \(P^{(i)} = \{n_1^{(i)}, \ldots, n_k^{(i)}\}.\) Accordingly, we define

\[\Sigma^{(i)} = \{x = (x_1^{(i)}, \ldots, x_{k_i}^{(i)})^\top \in \mathbb{R}^n : x_j \in \mathbb{R}^{n_j^{(i)}}, \|x_2\|_2 = 1, \text{ for } j = 1, 2, \ldots, k_i\}.\]

Several remarks are in order.

Firstly, any one of existing algorithms mentioned previously, i.e., the Horst algorithm, the Gauss-Seidel algorithm, the P-SOR algorithm, and the APM, can be chosen as the iteration algorithm \(\Phi.\) Both the Gauss-Seidel algorithm [11] and the P-SOR method [11] are proposed as natural improvements on the Horst method for the MEP (1.2). The Gauss-Seidel method is deduced from the P-SOR method by selecting all the relaxation parameters \(\omega^{(k)}_i = 1.\) An open question associated with the P-SOR method is how to establish the optimal relaxation parameters (if exist). Numerical testings made in [14] indicate that for some choices of \(\omega^{(k)}_i\), the convergence of the P-SOR is slower than the Gauss-Seidel algorithm.

For linear system of equations, an interesting observation shows that the convergence rate of the symmetric SOR (SSOR) is less sensitive to selection of the
Algorithm 1 A multigrid method for MCP

(1) Compute the solution of the following optimization problem
\[ \max_{x \in \Sigma} \rho(x). \] (4.1)

(2) For \( i = 2, 3, \cdots, L \)
(2.1) Solve
\[ \min_{x \in \Sigma} \|x^{(i-1)} - x\|_2, \] (4.2)
where \( x^{(i-1)} \) is the solution of the following problem
\[ \max_{x \in \Sigma} \rho(x). \]

(2.2) Let \( y^{(i)} \) be the solution of (4.2). Take \( y^{(i)} \) as the starting point to solve the following problem by some iteration method \( \Phi \):
\[ \max_{y \in \Sigma} \rho(y) \] (4.3)

end

relaxation parameter than the SOR iteration. See, e.g., [15, P118]. Therefore, we suggest that the APM and the symmetric Gauss-Seidel iteration are taken as candidates for the iteration \( \Phi \) in Algorithm 1. The framework of the APM has been outlined in [4] and [18]. The symmetric Gauss-Seidel (S-G-S) is summarized in Algorithm 2.

Algorithm 2 The framework of the symmetric Gauss-Seidel method (S-G-S)

Select \( x^{(0)} \in \Sigma_m \)
for \( k = 0, 1, \cdots \) do
  for \( i = 1, 2, \cdots, m \) do
    \[ y_i^{(k)} = \sum_{j=1}^{i-1} A_{ij} x_j^{(k+1)} + \sum_{j=i}^{m} A_{ij} x_j^{(k)}, \]
    \[ \mu_i^{(k)} = \|y_i^{(k)}\|_2, \]
    \[ \tilde{x}_i^{(k+1)} = y_i^{(k)} / \mu_i^{(k)}, \]
  end for
  for \( i = m, m - 1, \cdots, 1 \) do
    \[ z_i^{(k)} = \sum_{j=1}^{i} A_{ij} x_j^{(k+1)} + \sum_{j=i+1}^{m} A_{ij} x_j^{(k+1)}, \]
    \[ \lambda_i^{(k)} = \|z_i^{(k)}\|_2, \]
    \[ x_i^{(k+1)} = z_i^{(k)} / \lambda_i^{(k)}, \]
  end for
end for

Secondly, the case \( L = 2 \) is of special interest. For such a case, Algorithm 1 can be regarded as an algorithm \( \Phi \) for MCP with a specially selected starting point \( x^{(0)} \). Similar ideas have appeared in literature. For example, Zhang and Chu [17] propose the following starting point strategy for the Gauss-Seidel algorithm.

Strategy 1 Set \( v = (v_1^T, \cdots, v_m^T)^T \) as starting point, where \( v_i \) is the unit eigenvector associated with the largest eigenvalue of \( A_{ii} \).

Theorem 2.3 suggests that this is a good starting point selection. Numerical experiment in [17] indicates that this strategy boosts up the probability of finding
a global maximizer of (1.1). However, in some cases, e.g., $A_{ii} = I_{n_i}$, for $i = 1, 2, \cdots, m$, this strategy is no longer effective. Theorems 2.1 and Corollary 2.5 suggest the following starting point strategy.

**Strategy 2**

1. Compute the unit eigenvector $v \in \mathbb{R}^n$ associated with the largest eigenvalue of $A$.
2. Compute the projection $x^{(0)}$ of $v$ onto $\Sigma_m$, and take $x^{(0)}$ as starting point for an existing iteration algorithm (for example, the symmetric Gauss-Seidel algorithm) for MCP.

Finally, we point out that the following optimization can be regarded as a relaxation of the MCP (1.1):

$$\max_{x \in \Sigma^{(k)}} \rho(x), \ (1 \leq k \leq L - 1).$$

More information on the relaxation techniques for optimization problems can be found, for example, in [2, 16, 20].

5. **Numerical examples.** In this section, numerical experiments for the multigrid-type method are carried out to show its superior performance to the others in seeking a global maximizer of MCP. All of our tests were conducted in MATLAB 7.1. In the following numerical experiments, the choice of $\{\omega_i^{(k)}\}$ for the P-SOR algorithm is the same as Sun [11 eqn.(4.1)]. We stopped the iteration when $\|Ax^{(k)} - \Lambda^{(k)}x^{(k)}\|_2 \leq \epsilon$ is met for given tolerance $\epsilon$.

For comparison, we first carried out numerical tests on problems with known global maximizers in advance. Three small examples are available in the literature whose global maximizers have been obtained through an exhaustive search. For reference, we list these examples below.

**Example 5.1.** [1] The matrix $A$ is given by

$$A = \begin{pmatrix}
4.3229 & 2.3230 & -1.3711 & -0.0084 & -0.7414 \\
3.2320 & 3.1181 & 1.0959 & 0.1285 & 0.0727 \\
-1.3711 & 1.0959 & 6.4920 & -1.9883 & -0.1878 \\
-0.0084 & 0.1285 & -1.9883 & 2.4591 & 1.8463 \\
-0.7414 & 0.0727 & -0.1878 & 1.8463 & 5.8875
\end{pmatrix},$$

with $m = 2$ and $P = \{2, 3\}$.

**Example 5.2.** [14] The matrix $A$ is given by

$$A = \begin{pmatrix}
A_{11} & A_{12} & A_{13} \\
A_{12} & A_{22} & A_{23} \\
A_{13} & A_{23} & A_{33}
\end{pmatrix},$$

$$A_{11} = \begin{pmatrix}
7.4470 & -5.3387 & -4.8906 \\
-5.3387 & 13.8905 & 4.5704 \\
-4.8906 & 4.5704 & 8.8486
\end{pmatrix},$$

$$A_{12} = \begin{pmatrix}
0.0903 & 2.2446 & 3.6421 \\
-1.4263 & 0.5230 & 1.8020 \\
-0.1523 & -0.3143 & -1.6288
\end{pmatrix},$$
\[ A_{13} = \begin{pmatrix} -0.1752 & -0.0246 & 1.7437 & 5.8028 \\ -3.7392 & 3.5696 & -1.9936 & -8.1650 \\ -1.3250 & 0.4012 & -2.7730 & -2.6152 \end{pmatrix}, \]

\[ A_{22} = \begin{pmatrix} 4.4653 & -4.2442 & -5.9271 \\ -4.2442 & 25.9532 & 6.8747 \\ -5.9271 & 6.8747 & 12.3494 \end{pmatrix}, \]

\[ A_{23} = \begin{pmatrix} -3.8049 & -3.9631 & 5.1445 & -0.4010 \\ -4.8952 & -0.8508 & 3.9045 & -4.8526 \\ 2.8848 & 3.0664 & -3.8247 & 0.7834 \end{pmatrix}, \]

\[ A_{33} = \begin{pmatrix} 11.7609 & 5.7035 & -7.7411 & 6.0963 \\ 5.7035 & 12.4512 & -11.9284 & 4.7289 \\ -7.7411 & -11.9284 & 16.8287 & -5.8446 \\ 6.0963 & 4.7289 & -5.8446 & 11.6403 \end{pmatrix}, \]

with \( m = 3 \) and \( \mathcal{P} = \{3, 3, 4\} \).

**Example 5.3.** [6] The matrix \( A \) is given by

\[ A = \begin{pmatrix} 45 & -20 & 5 & 6 & 16 & 3 \\ -20 & 77 & -20 & -25 & -8 & -21 \\ 5 & -20 & 74 & 47 & 18 & -32 \\ 6 & -25 & 47 & 54 & 7 & -11 \\ 16 & -8 & 18 & 7 & 21 & -7 \\ 3 & -21 & -32 & -11 & -7 & 70 \end{pmatrix}, \]

with \( m = 3 \) and \( \mathcal{P} = \{2, 2, 2\} \).

For comparison, we ran five algorithms for these examples: the APM, the Horst algorithm, the Gauss-Seidel (G-S) algorithm, the symmetric Gauss-Seidel (S-G-S) algorithm and the P-SOR algorithm.

We ran all algorithms starting from \( 10^4 \) random initial points, and used \( \epsilon = 10^{-6} \) as the tolerance. The numerical results are displayed in Table 1. Under the columns “Avg.Iter.#” are the average numbers of iterations needed to meet the control precision. Under columns “% to global” are the sample probabilities, out of the \( 10^4 \) random tests, of convergence to a global maximizer of MCP.

| Method | Example 5.1 | Example 5.2 | Example 5.3 |
|--------|-------------|-------------|-------------|
|        | Avg.Iter.#  | % to global | Avg.Iter.#  | % to global | Avg.Iter.#  | % to global |
| Horst  | 67.12       | 53.66       | 27.39       | 28.34       | 20.22       | 72.53       |
| G-S    | 63.74       | 54.03       | 25.21       | 28.86       | 19.50       | 73.64       |
| P-SOR  | 35.11       | 54.23       | 22.64       | 28.87       | 20.06       | 75.59       |
| S-G-S  | 34.59       | 54.58       | 14.12       | 29.03       | 10.72       | 73.78       |
| APM    | 6.34        | 91.60       | 8.01        | 72.54       | 5.13        | 100         |

Table 1. Performance of the five algorithms on Examples 5.1, 5.2 and 5.3.
The five algorithms all succeed in finding a solution of MEP for all $10^4$ random starting points. The results in Table 1 indicate that the APM has the fewest iteration numbers and largest probability to reach a global maximizer of MCP among the five methods. The results with Example 5.1 and Example 5.2 show that selecting starting point randomly, though the APM has a larger probability to reach a global maximizer than others, it cannot guarantee to always converge to a global optimal solution. At the same time, our numerical experiments with the Algorithm 1 obtained a global maximizer of MCP for each of these examples.

We ran four algorithms with various starting point strategies, and the numerical results are displayed in Table 2. Under the columns “Iter” are the numbers of iterations needed to meet the control precision. Under columns “Y/N” (Yes/No) represent whether the algorithm converged to a global optimal solution of MCP or not and “MG” represents Multigrid method.

**Table 2.** Performance of the four algorithms with specially chosen starting point on Examples 5.1, 5.2 and 5.3

| Example 5.1 | Example 5.2 | Example 5.3 |
|-------------|-------------|-------------|
| Method      | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    | Iter Y/N    |
| Horst       | 50 Y        | 28 N        | 14 Y        | 12 Y        | Y           | Y           |
| P-SOR       | 27 Y        | 18 N        | 18 Y        | 18 Y        | Y           | Y           |
| S-G-S       | 26 Y        | 13 N        | 8 Y         | 7 Y         | Y           | Y           |
| APM         | 6 Y         | 8 Y         | 5 Y         | 4 Y         | Y           | Y           |

The results in Table 2 clearly demonstrate the power of some starting point strategies in boosting up the probability of finding a global maximizer of MCP, and what we are interested in here is that the multigrid-method combined with any one of the four known algorithms always converge to a global optimal solution of MCP. We carried out many further experiments and found that the multigrid-method always reached a maximizer of MCP.

**Example 5.4.** This example is created as follows: first, an $N$-by-$n$ ($N > n$ and let $n = 50$ and 100 respectively) matrix $Y$ with rank($Y$) = $n$ is generated randomly with MATLAB 7.1. Then we let $A = Y^\top Y$. Thus, $A$ is an $n$-by-$n$ symmetric and positive definite matrix. In the third step, we partition $A$ into block forms according to various $m$ and $P$, where $m$ and $P$ are left to our decision.

Numerical results are recorded in Table 3. Under the column “$\rho$” are the maxima of MCP. These experiment results seem to suggest that (i) the P-SOR method shows no advantage on larger scale problems, (ii) $\rho$ increases with the increase of $m$.

**Acknowledgments.** The authors would like to thank the two referees for their invaluable comments and suggestions which improved the presentation of the paper.
### Table 3. Performance of the five algorithms on Example 5.4

| Method | m(n = 50) | Iter. | ρ   | m(n = 100) | Iter. | ρ   |
|--------|-----------|-------|-----|------------|-------|-----|
| Horst  | 3         | 4     | 3.734e+003 | 5     | 3.734e+003 | 3.409e+003 |
|        | 5         | 5     | 6.247e+003 | 5     | 6.247e+003 | 2.509e+004 |
|        | 7         | 5     | 8.512e+003 | 7     | 8.512e+003 | 3.470e+004 |
| P-SOR  | 3         | 28    | 3.734e+003 | 5     | 3.734e+003 | 3.409e+003 |
|        | 5         | 29    | 6.247e+003 | 5     | 6.247e+003 | 2.509e+004 |
|        | 7         | 29    | 8.512e+003 | 7     | 8.512e+003 | 3.470e+004 |
| S-G-S  | 3         | 3     | 3.734e+003 | 5     | 3.734e+003 | 3.409e+003 |
|        | 5         | 5     | 6.247e+003 | 5     | 6.247e+003 | 2.509e+004 |
|        | 7         | 5     | 8.512e+003 | 7     | 8.512e+003 | 3.470e+004 |
| APM    | 5         | 4     | 6.247e+003 | 5     | 6.247e+003 | 2.509e+004 |
|        | 7         | 4     | 8.512e+003 | 7     | 8.512e+003 | 3.470e+004 |
| Multigrid | 3     | 2     | 3.734e+003 | 5     | 3.734e+003 | 3.409e+003 |
|        | 5         | 4     | 6.247e+003 | 5     | 6.247e+003 | 2.509e+004 |
|        | 7         | 6     | 8.512e+003 | 7     | 8.512e+003 | 3.470e+004 |

**REFERENCES**

[1] M. T. Chu and J. L. Watterson, *On a multivariate eigenvalue problem, Part I: Algebraic theory and a power method*, SIAM J. Sci. Comput., 14 (1993), 1089–1106.

[2] M. Y. Fu, Z. Q. Luo and Y. Y. Ye, *Approximation algorithms for quadratic programming*, J. Comb. Optim., 2 (1998), 29–50.

[3] G. H. Golub and C. F. Van Loan, “Matrix Computations,” Third Edition, The Johns Hopkins University Press, Baltimore, 1996.

[4] S. M. Grzegórski, *On the convergence of the method of alternating projections for multivariate symmetric eigenvalue problem*, Numan 2010, Conference in Numerical Analysis, Chania, Greece, Sept 15-18, 2010.

[5] W. Hackbusch, “Multi-grid Method and Applications,” Springer-Verlag, New York, 1985.

[6] M. Hanafi and J. M. F. Ten Berge, *Global optimality of the successive Maxbet algorithm*, Psychometrika, 68 (2003), 97–103.

[7] P. Horst, *Relations among m sets of measures*, Psychometrika, 26 (1961), 129–149.

[8] D.-K. Hu, *The convergence property of a algorithm about generalized eigenvalue and eigenvector of positive definite matrix*, China-Japan Symposium on Statistics, 1984, Beijing, China.

[9] J. R. Kettenring, *Canonical analysis of several sets of variables*, Biometrika, 58 (1971), 433–451.

[10] Z.-Y. Liu, J. Qian and S.-F. Xu, *On the double eigenvalue problem*, preprint. Available online: [http://www.math.pku.edu.cn:8000/var/preprint/7229.pdf](http://www.math.pku.edu.cn:8000/var/preprint/7229.pdf).

[11] J.-G. Sun, *An algorithm for the solution of multiparameter eigenvalue problem*, Math. Numer. Sinica(Chinese), 8 (1986), 137–149.

[12] J. M. F. Ten Berge, *Generalized approaches to the MAXBET problem and the MAXDIFF problem, with applications to canonical correlations*, Psychometrika, 53 (1988), 487–494.

[13] T. L. Van Noorden and J. Barkmeijer, *The multivariate eigenvalue problem: A new application, theory and a subspace accelerated power method*, Universiteit Utrecht, preprint, 2008. Available online: [http://www.math.uu.nl/publications/preprints/1308.ps](http://www.math.uu.nl/publications/preprints/1308.ps).

[14] L.-H. Xu, *Numerical Methods for the Multivariate Eigenvalue Problem*, M.S. Thesis, Department of Mathematics, Ocean University of China, 2008, ( in Chinese). Available from: [http://cdmd.cnki.com.cn/Article/CDMD-10423-2008175406.htm](http://cdmd.cnki.com.cn/Article/CDMD-10423-2008175406.htm).

[15] S.-F. Xu, “Matrix Computations: Theory and Methods,” Peking University Press, Beijing, 1995 (Chinese).

[16] Y. Y. Ye, *Approximating quadratic programming with bound and quadratic constraints*, Math. Program., 84 (1999), 219–226.

[17] L.-H. Zhang and M. T. Chu, *Computing absolute maximum correlation*, IMA J. Numer. Anal., 32 (2012), 163–184.

[18] L.-H. Zhang and L.-Z. Liao, *An alternating variable method for the maximal correlation problem*, J. Global Optim., 54 (2012), 199–218.
[19] L.-H. Zhang, L.-Z. Liao and L.-M. Sun, *Towards the global solution of the maximal correlation problem*, J. Global Optim., 49 (2011), 91–107.

[20] L. Zhang, Y. Xu and Z. Jin, *An efficient algorithm for convex quadratic semi-definite optimization*, Numer. Algebra Control Optim., 2 (2012), 129–144.

Received December 2011; 1st revision March 2012; 2nd revision September 2012.

*E-mail address*:

liuxinguo@ouc.edu.cn

E-mail address:

sdtawk@sina.com