
Volume 37
Issue 6
June 2010
ISSN 0305-0548
ELSEVIER
computers & operations research
Editor: Stefan Nickel
Available online at www.sciencedirect.com
ScienceDirect

Computers & Operations Research 37 (2010) 1027–1032

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

An efficient continuation method for quadratic assignment problems ☆

Yong Xia

LMIB of the Ministry of Education, School of Mathematics and System Sciences, Beihang University, Beijing 100191, PR China

ARTICLE INFO

Available online 15 September 2009

Keywords:
Quadratic assignment problem
Lagrangian smoothing
Frank–Wolfe algorithm
KKT
Heuristic

ABSTRACT

In this article, we propose a Lagrangian smoothing algorithm for quadratic assignment problems, where the continuation subproblems are solved by the truncated Frank–Wolfe algorithm. We establish practical stopping criteria and show the algorithm finitely terminates at a KKT point of a continuation subproblem. The quality of the returned solution is studied in detail. Finally, limited numerical results are provided.

1. Introduction

Quadratic assignment problem (QAP) is one of the great challenges in combinatorial optimization. It is known to be NP-hard. An ε-solution is NP-hard too. For comprehensive surveys of QAPs, we refer to [1,8,9,11,14]. The formulation of QAP can be written as

$$\begin{aligned}\min\quad & f(X) = \text{trace }(AXBX^T)\\ \text{s.t.}\quad & X \in \Pi_n,\end{aligned} \tag{1.1}$$

where A and B are $n \times n$ matrices, 'trace' denotes the sum of all diagonal elements, and Π_n is the set of $n \times n$ permutation matrices, i.e., $\Pi_n = \{X = (X_{ij}) \in \mathbb{R}^{n\times n} : Xe = X^Te = e, X_{ij} \in \{0,1\}\}$, where e is a vector with all components equal to one. Denote the continuous relaxation of Π_n by $P_n = \{X = (X_{ij}) \in \mathbb{R}^{n\times n} :\ Xe = X^Te = e, X_{ij} \geq 0\}$, which is also the convex hull of Π_n, i.e., $P_n = \text{conv}\{\Pi_n\}$.

In most practical applications, the QAP models are symmetric, i.e., both A and B are symmetric. Furthermore, if only one of these matrices is symmetric (say A), we can transform it to a QAP where both matrices are symmetric since

$$\text{trace }(AXB^TX^T) = \text{trace }\left(AX\frac{B^T+B}{2}X^T\right).$$

Therefore, we make the symmetric assumption throughout this article.

Continuation methods are new promising approaches to solving QAP. In [12] the logarithmic smoothing algorithm (LogSA) was applied to solve QAP and limited numerical results were reported. The Lagrangian smoothing algorithm for QAP was firstly devised in [16]. Numerical experiments indicated its high efficiency. In this article, we further study this algorithm. We propose a new version, establish practical stopping criteria and prove the algorithm finitely terminates at a KKT point of a continuation subproblem. We also analyze the quality of the obtained solution.

The article is organized as follows. In Section 2, a new version of Lagrangian smoothing algorithm is proposed. The convergence result is analyzed in Section 3. Implementation details and numerical results are presented in Section 4. The last section makes some concluding remarks.

Notation. For a square matrix A, vec(A) gives the column vector obtained by stacking the columns of A in increasing order of their index. The Kronecker product of matrices A and B is denoted by $A \otimes B$. Let S^n be the set of symmetric matrices of order n, i.e., $S^n = \{X \in \mathbb{R}^{n\times n} : X^T = X\}$. For $A, B \in S^n$, $A \succcurlyeq B$ denotes that $A - B$ is positive semidefinite. $\lambda_{\min}(A)$ denotes the minimal eigenvalue of A. Let C^n be the set of functions with continuous n-th order derivatives. ∇f and $\nabla^2 f$ denote the gradient and Hessian of f, respectively.

2. Lagrangian smoothing algorithm

Basically, global smoothing is to linearly combine the original objective $f(X)$ with an additional function $\Phi(X)$ such that

$$F(X;\mu) = f(X) + \mu \cdot \Phi(X) \tag{2.1}$$

is strictly convex on P_n with respect to X.

Theorem 2.1 (*Bertsekas [3] and Ng [12]*). *Suppose $\Phi : P_n \to \mathbb{R}$ is a C^2 function such that the minimum eigenvalue of $\nabla^2\Phi$ is greater than a positive number ε for all $X \in P_n$. Then there exists a real $M > 0$ such that if $\mu > M$, then $f + \mu \cdot \Phi$ is a strictly convex function on P_n.*

☆ Supported by Chinese NSF Grant 10671199.
E-mail address: dearyxia@gmail.com

0305-0548/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2009.09.002

The logarithmic barrier function

$$\Phi^{Log}(X) = -\sum_{i=1}^{n}\sum_{j=1}^{n}\ln(X_{ij}) - \sum_{i=1}^{n}\sum_{j=1}^{n}\ln(1-X_{ij}) \quad (2.2)$$

was introduced as a smoothing function in [12]. It is well-defined when $X\in(0,1)^{n\times n}$. If any value of X_{ij} is 0 or 1, then $\Phi^{Log}(X)=+\infty$.

Theorem 2.2 (*Ng [12]*). *There exists a real $M>0$ such that if $\mu\ge M$, then $f+\mu\cdot\Phi^{Log}$ is a strictly convex function on $(0,1)^{n\times n}$.*

Notice that QAP (1.1) is equivalent to

$$\min \quad \text{trace } (AXBX^T) \quad (2.3)$$
$$\text{s.t.} \quad \text{trace } (XX^T)=n, \quad (2.4)$$
$$X\in P_n. \quad (2.5)$$

Introducing a Lagrangian multiplier for the constraint (2.4), we obtain the corresponding Lagrangian function

$$L(X;\mu_0)=\text{trace } (AXBX^T)+\mu_0\cdot\text{trace } (XX^T)-n\mu_0, \quad (2.6)$$

which motivates the following global smoothing function [16]:

$$\Phi^{Lag}(X)=\text{trace } (XX^T). \quad (2.7)$$

Theorem 2.3. *Let the $n\times(n-1)$ matrix V be such that*

$$V^Te=0, \quad V^TV=I_{n-1}.$$

Define $\hat{A}=V^TAV$, $\hat{B}=V^TBV$ and

$$\hat{\lambda}_{\min}=\min\{\lambda_{\min}(\hat{B})\lambda_{\max}(\hat{A}),\lambda_{\max}(\hat{B})\lambda_{\min}(\hat{A}),\lambda_{\min}(\hat{B})\lambda_{\min}(\hat{A}),\lambda_{\max}(\hat{B})\lambda_{\max}(\hat{A})\}. \quad (2.8)$$

Then $f+\mu_0\cdot\Phi^{Lag}$ is a strictly convex function on P_n for any $\mu_0>-\hat{\lambda}_{\min}$.

Proof. Let

$$Q=\left[\frac{e}{\sqrt{n}}\vdots V\right]\in O_n := \{X\in\mathbb{R}^{n\times n}: XX^T=I_n\}.$$

We introduce a well-known result due to Hadley et al. [10].

Lemma 2.1 (*Hadley et al. [10]*). *Let X be $n\times n$ and Y be $(n-1)\times(n-1)$. Suppose that X and Y satisfy*

$$X=Q\begin{bmatrix}1&0\\0&Y\end{bmatrix}Q^T=\frac{1}{n}ee^T+VYV^T. \quad (2.9)$$

Then

$$Xe=X^Te=e,$$
$$X\in O_n \Longleftrightarrow Y\in O_{n-1}.$$

Taking Eq. (2.9) into $f(X)$ and $\Phi^{Lag}(X)$, respectively, we have

$$\text{trace } (AXBX^T)=\text{trace } (\hat{A}Y\hat{B}Y^T)+\frac{2}{n}\text{trace } (V^TAee^TBVY^T)+\frac{1}{n^2}(e^TAe)(e^TBe),$$
$$\text{trace } (XX^T)=\text{trace } (YY^T)+1.$$

Consequently, the reduced Hessian of $f+\mu_0\cdot\Phi^{Lag}$ is $2\hat{B}\otimes\hat{A}+2\mu_0I$. It is positive definite if and only if $\mu_0>-\lambda_{\min}(\hat{B}\otimes\hat{A})=-\hat{\lambda}_{\min}$, where $\hat{\lambda}_{\min}$ is defined in (2.8). □

Linearly combining $f+\mu_0\cdot\Phi^{Lag}$ with the exact penalty function (see Theorem 2.4):

$$P(X;\mu_\infty)=\text{trace } (AXBX^T)+\mu_\infty\cdot\text{trace } (XX^T), \quad (2.10)$$

we obtain a sequence of functions

$$PL(X;\mu)=\text{trace } (AXBX^T+\mu XX^T), \quad (2.11)$$

and define

$$H(X;\mu)=AXBX^T+\mu XX^T. \quad (2.12)$$

Then we solve the following parametric optimization problem, denoted by $QAP(\mu)$:

$$\min \quad PL(X;\mu)$$
$$\text{s.t.} \quad X\in P_n, \quad (2.13)$$

for a decreased sequence $\{\mu\}\subseteq[\mu_\infty,\mu_0]$.

Theorem 2.4. *The optimal function values of QAP (1.1) and Problem (2.13) are equal for all $\mu=\mu_\infty\le-\hat{\lambda}_{\max}$, where*

$$\hat{\lambda}_{\max}=\max\{\lambda_{\min}(\hat{B})\lambda_{\max}(\hat{A}),\lambda_{\max}(\hat{B})\lambda_{\min}(\hat{A}),\lambda_{\min}(\hat{B})\lambda_{\min}(\hat{A}),\lambda_{\max}(\hat{B})\lambda_{\max}(\hat{A})\}. \quad (2.14)$$

Proof. Since $P_n=\text{conv}\{\Pi_n\}$ and Π_n is the extreme point set of P_n, it is sufficient to show that $PL(X;\mu)$ is concave on P_n for all $\mu\le-\hat{\lambda}_{\max}$. Similarly to the proof of Theorem 2.3, we conclude that the reduced Hessian of $PL(X;\mu)$ is $2\hat{B}\otimes\hat{A}+2\mu I$ and it is negative semidefinite if and only if $\mu\le-\lambda_{\max}(\hat{B}\otimes\hat{A})=-\hat{\lambda}_{\max}$, where $\hat{\lambda}_{\max}$ is defined in (2.14). □

Given any parameter μ, we use the canonical Frank–Wolfe algorithm [5] to solve the subproblem (2.13). It approximates the objective function with its first order Taylor expansion at any given iteration point X_k, resulting in the linear programming subproblem (omitting the constant terms)

$$\min \quad \text{trace } (\nabla_x H(X_k;\mu)\cdot X^T)$$
$$\text{s.t.} \quad X\in P_n, \quad (2.15)$$

where $\nabla_x H(X_k;\mu)\in\mathbb{R}^{n\times n}$ denotes the gradient of $H(X;\mu)$ with respect to X at X_k, i.e.,

$$\nabla_x H(X_k;\mu)=2AX_kB+2\mu X_k.$$

Furthermore, Problem (2.15) is a typical linear assignment problem (LAP) and can be solved in $O(n^3)$ time, for example, using Hungarian method.

The optimal solution of the LAP (2.15), X_k^*, is used to construct the descent search direction $D_k=X_k^*-X_k$. A line search

$$\alpha^*=\arg\min_{\alpha\in[0,1]} PL(X_k+\alpha D_k;\mu) \quad (2.16)$$

furnishes the next iteration

$$X_{k+1}=X_k+(1-\beta+\beta\alpha^*)\alpha^*D_k, \quad (2.17)$$

where $\beta\in[0,1]$ is a fixed parameter and the process is repeated. Generally, the new step length satisfies $(1-\beta+\beta\alpha^*)\alpha^*\le\alpha^*$, where the equality holds when $\beta=0$ or $\alpha^*=1$.

It is easy to verify that the point sequence $\{X_k\}$ generated by the above Frank–Wolfe algorithm converges to X^*, a KKT point of (2.13). But the convergence is slow and hence it is quite time-consuming to obtain X^*. This phenomenon motivates us to approximate X^* using truncated Frank–Wolfe algorithm, which only generates the first several iterative points.

The above process is repeated for a decreased sequence $\{\mu_i\}$, the so-called outer iteration.

To introduce practical stopping criteria, we need some lemmas.

Lemma 2.2. *Suppose $X_k\in\Pi_n$ is an optimal solution of the LAP (2.15), i.e., trace $(\nabla_x H(X_k;\mu)(X_k-X_k^*)^T)=0$. Then it is also a KKT point of $QAP(\mu)$ (2.13).*

Proof. The KKT system of $QAP(\mu)$ (2.13) is

$$(\nabla_x H(X;\mu))_{ij}-\lambda_i-\sigma_j\ge 0, \quad i,j=1,\ldots,n, \quad (2.18)$$

$$X_{ij}((\nabla_x H(X;\mu))_{ij}-\lambda_i-\sigma_j)=0, \quad i,j=1,\ldots,n, \quad (2.19)$$

$$X\in P_n. \quad (2.20)$$

Since X_k is an optimal solution of the LAP (2.15) and the latter is a linear programming with bounded feasible region, it holds that

$$(\nabla_x H(X_k;\mu))_{ij} - \lambda_i - \sigma_j \geq 0, \quad i,j = 1,\dots,n, \tag{2.21}$$

$$(X_k)_{ij}((\nabla_x H(X_k;\mu))_{ij} - \lambda_i - \sigma_j) = 0, \quad i,j = 1,\dots,n, \tag{2.22}$$

$$X_k \in P_n, \tag{2.23}$$

which completes the proof. □

Lemma 2.3. *If $\hat{X} \in \Pi_n$ is a KKT point of QAP(μ) (2.13), then it remains a KKT point of QAP(μ') for all $\mu' < \mu$.*

Proof. $\hat{X} \in \Pi_n$ is a KKT point of $QAP(\mu)$, which implies that the KKT conditions (2.18)–(2.20) hold for $\hat{X}$ and some (λ, σ). Define

$$\hat{\lambda} = \lambda - (\mu - \mu')e < \lambda, \tag{2.24}$$

$$\hat{\sigma} = \sigma - (\mu - \mu')e < \sigma. \tag{2.25}$$

Then $(\hat{X}, \hat{\lambda}, \hat{\sigma})$ satisfies the KKT conditions of $QAP(\mu')$ because

$$(\nabla_x H(X;\mu'))_{ij} - \hat{\lambda}_i - \hat{\sigma}_j = (\nabla_x H(X;\mu))_{ij} - \lambda_i - \lambda_j + 2(\mu - \mu')(1 - X_{ij}) \geq 0,$$

$$X_{ij}((\nabla_x H(X;\mu'))_{ij} - \hat{\lambda}_i - \hat{\sigma}_j) = X_{ij}((\nabla_x H(X;\mu))_{ij} - \lambda_i - \sigma_j) + 2(\mu - \mu')(X_{ij} - X_{ij}^2) = 0,$$

for $i,j = 1,\dots,n$. □

Notice that if a KKT point is obtained in Π_n, it is not needed to update μ due to Lemma 2.3. We take the condition that $X_k \in \Pi_n$ is optimal to the LAP (2.15) as stopping criteria.

The detailed algorithm can be formally proposed as follows, denoted by LagSA.

Algorithm 2.1 (*LagSA*). *Step 1*: Initialize $i := 0$, $X_0 \in P_n$. Set $\beta \in [0,1]$, m and $\mu_0 \geq -\hat{\lambda}_{\min}$.

Step 2: For $k := 0$ to $m-1$, compute $X_k^* \in \Pi_n$ and α^* according to (2.15) and (2.16) respectively; update

$$X_{k+1} := X_k + (1 - \beta + \beta\alpha^*)\alpha^*(X_k^* - X_k); \tag{2.26}$$

if $X_k \in \Pi_n$ and trace $(\nabla_x H(X_k;\mu_i)(X_k - X_k^*)^T) = 0$, then goto Step 4, otherwise goto Step 3.

Step 3: Choose $\mu_{i+1} < \mu_i$, update $i := i+1$, $X_0 := X_m$, and goto Step 2.

Step 4: Stop and return $X^* = X_k$.

3. Convergence analysis

Lemma 3.1. *Suppose*

$$\mu \leq \min_{i,j} \min_{X \in \Pi_n} (AXB)_{ij} - \max_{i,j} \max_{X \in \Pi_n} (AXB)_{ij}. \tag{3.1}$$

Then any $X \in \Pi_n$ is a KKT point of QAP(μ) (2.13).

Proof. Let $X \in \Pi_n$ and the corresponding permutation π is defined by $\pi(i) = j$ where $X_{ij} = 1$. The KKT conditions (2.18)–(2.20) are equivalent to

$$(\nabla_x H(X;\mu))_{ij} - \lambda_i - \sigma_j \geq 0, \quad \forall i, j \neq \pi(i), \tag{3.2}$$

$$(\nabla_x H(X;\mu))_{i\pi(i)} - \lambda_i - \sigma_{\pi(i)} = 0, \quad \forall i, \tag{3.3}$$

$$X \in P_n. \tag{3.4}$$

We notice that

$$(\nabla_x H(X;\mu))_{ij} = 2(AXB)_{ij}, \quad \forall i, j \neq \pi(i), \tag{3.5}$$

$$(\nabla_x H(X;\mu))_{i\pi(i)} = 2(AXB)_{i\pi(i)} + 2\mu, \quad \forall i. \tag{3.6}$$

For $i = 1,\dots,n$, set

$$\lambda_i = \tfrac{1}{2}(\nabla_x H(X;\mu))_{i\pi(i)}, \tag{3.7}$$

$$\sigma_{\pi(i)} = \tfrac{1}{2}(\nabla_x H(X;\mu))_{i\pi(i)}. \tag{3.8}$$

Eq. (3.3) is satisfied. It is sufficient to show (3.7) and (3.8) satisfy (3.2), i.e., for all i and $j \neq \pi(i)$,

$$\mu \leq (AXB)_{ij} - (AXB)_{i\pi(i)}. \tag{3.9}$$

It holds true if we choose μ satisfying

$$\mu \leq \min_{X \in \Pi_n} \min_{i,j} (AXB)_{ij} - \max_{X \in \Pi_n} \max_{i,j} (AXB)_{ij} = \min_{i,j} \min_{X \in \Pi_n} (AXB)_{ij} - \max_{i,j} \max_{X \in \Pi_n} (AXB)_{ij}. \quad \square \tag{3.10}$$

Lemma 3.2. *Consider applying Frank–Wolfe algorithm to solve QAP(μ) (2.13) with $\mu < -\hat{\lambda}_{\max}$ where $\hat{\lambda}_{\max}$ is defined in (2.14). Let $X_0 \in P_n$ be any given initial point and X_1 be the next generated iterative point. Then $X_1 \in \Pi_n$.*

Proof. The condition $\mu < -\hat{\lambda}_{\max}$ implies that $\hat{B} \otimes \hat{A} + \mu I$, the reduced Hessian of $PL(X;\mu)$ is negative definite. That is, $PL(X;\mu)$ is strictly concave on P_n.

Applying Frank–Wolfe algorithm to solve $QAP(\mu)$, we first solve the LAP (see also (2.15))

$$\begin{aligned} \min \quad & \text{trace } (\nabla_x H(X_0;\mu) X^T) \\ \text{s.t.} \quad & X \in P_n, \end{aligned} \tag{3.11}$$

and get an optimal solution $X_0^* \in \Pi_n$. Since X_0 is feasible in (3.11), we have

$$\text{trace } (\nabla_x H(X_0;\mu)(X_0^* - X_0)^T) \leq 0. \tag{3.12}$$

Then $D_0 = X_0^* - X_0$ is a descent direction at X_0. We do the line search

$$\alpha^* = \arg\min_{\alpha \in [0,1]} PL(X_0 + \alpha D_0;\mu) \tag{3.13}$$

and update

$$X_1 = X_0 + (1 - \beta + \beta\alpha^*)\alpha^* D_0 = X_0 + (1 - \beta + \beta\alpha^*)\alpha^*(X_0^* - X_0). \tag{3.14}$$

To complete the proof, it is sufficient to show $\alpha^* = 1$ is the strict global optimal solution of (3.13) in the nontrivial case of $X_0 \neq X_0^*$. (Note that $0 \leq 1 - \beta + \beta\alpha^* \leq 1$.)

Since $PL(X + \alpha D_0;\mu)$ is a quadratic function of X,

$$\begin{aligned} PL(X_0^*;\mu) &= PL(X_0;\mu) + \text{trace } (\nabla_x H(X_0;\mu)(X_0^* - X_0)^T) \\ &\quad + \tfrac{1}{2}\text{vec}(X_0^* - X_0)^T(B \otimes A + \mu I)\text{vec}(X_0^* - X_0) \\ &\leq PL(X_0;\mu) + \tfrac{1}{2}\text{vec}(X_0^* - X_0)^T(B \otimes A + \mu I)\text{vec}(X_0^* - X_0) \\ &= PL(X_0;\mu) + \tfrac{1}{2}\text{vec}(Y_0^* - Y_0)^T(\hat{B} \otimes \hat{A} + \mu I)\text{vec}(Y_0^* - Y_0) \\ &< PL(X_0;\mu), \end{aligned}$$

where the first inequality follows from (3.12) and in the second equality the transformation (2.9) is applied to X_0^* and X_0, respectively.

Therefore, for all $0 \leq \alpha < 1$,

$$\begin{aligned} PL(X_0 + \alpha D_0;\mu) &= PL((1-\alpha)X_0 + \alpha X_0^*;\mu) \\ &\geq (1-\alpha)PL(X_0;\mu) + \alpha PL(X_0^*;\mu) \\ &> PL(X_0^*;\mu), \end{aligned}$$

where the first inequality follows from the strict concavity of $PL(X;\mu)$ on P_n. □

Theorem 3.1. *Let the maximum number of inner iterations be a fixed positive integer and the smoothing and penalty parameter sequence $\{\mu_i\}$ satisfies $\lim_{i\to\infty}\mu_i = -\infty$. Then Algorithm LagSA finitely terminates at a KKT point of QAP(μ_i) (2.13) for some i.*

Proof. Since $\lim_{i\to\infty}\mu_i = -\infty$, there exists a finite index I such that the following inequality holds for all $i > I$:

$$\mu_i < \min\left\{\min_{i,j}\min_{X\in\Pi_n}(AXB)_{ij} - \max_{i,j}\max_{X\in\Pi_n}(AXB)_{ij}, -\hat{\lambda}_{\max}\right\}. \tag{3.15}$$

Therefore, Lemmas 3.1 and 3.2 hold true. According to Lemma 3.2, after one Frank–Wolfe iteration, we obtain an $X^* \in \Pi_n$, which is a KKT point of $QAP(\mu_i)$ due to Lemma 3.1. □

Theorem 3.2. *Suppose Algorithm LagSA stops in the first outer iteration (i.e., i remains 0), then the returned X^* is also a globally optimal solution for QAP* (1.1).

Proof. If Algorithm LagSA stops in the first outer iteration and returns X^*, then $X^* \in \Pi_n$ is a KKT point of $QAP(\mu_0)$ (Lemma 2.2). We notice that $QAP(\mu_0)$ is a convex quadratic programming. Therefore, the KKT point X^* is also optimal to $QAP(\mu_0)$. That is,

$$\begin{aligned}
&PL(X^*;\mu_0) \le PL(X;\mu_0), \forall X\in\Pi_n\\
&\Leftrightarrow \text{trace } (AX^*BX^{*T}) + \mu_0\cdot \text{trace } X^*X^{*T}\\
&\quad\le \text{trace } (AXBX^T)+\mu_0\cdot\text{trace } XX^T, \quad \forall X\in\Pi_n\\
&\Leftrightarrow \text{trace } (AX^*BX^{*T}) \le \text{trace } (AXBX^T), \quad \forall X\in\Pi_n,
\end{aligned}$$

where the equalities trace $(X^*X^{*T}) = \text{trace } (XX^T) = n$ are used. The last inequalities imply that X^* is also a globally optimal solution for QAP (1.1). □

To further study the quality of X^* obtained by running Algorithm LagSA, we first introduce a class of locally optimal solutions of QAP.

Definition 3.1. $X^*\in\Pi_n$ is a q-surface solution of QAP ($q = 1, 2, \ldots, n$), if

$$\text{trace } (AX^*BX^{*T}) \le \min \quad \text{trace } (AXBX^T) \tag{3.16}$$

$$\text{s.t.} \quad X\in\Pi_n^q := \{X\in\Pi_n : \text{trace } (X^TX^*) = n-q\}. \tag{3.17}$$

Remark 3.1. It is easy to verify that the set of all possible values of trace (X^*X^T) is $\{0,1,\ldots,n\}$. Therefore, we have the decomposition $\Pi_n = \bigcup_{q=0,1,\ldots,n}\Pi_n^q$, where $\Pi_n^0 = \{X^*\}$.

Theorem 3.3. *Suppose Algorithm LagSA stops in the i-th outer iteration and returns a solution X^*. If it holds for some $q\in\{1,2,\ldots,n\}$ that*

$$\mu_i \ge \frac{u_q}{q} - \frac{n}{q}\hat{\lambda}_{\min}, \tag{3.18}$$

where $\hat{\lambda}_{\min}$ is defined in (2.8) *and u_q is obtained by solving the following convex quadratic programming, i.e.,*

$$u_q := \max \quad \text{trace } (A(X^*-X)BX^T + \hat{\lambda}_{\min}XX^T) \tag{3.19}$$

$$\text{s.t.} \quad \text{trace } (X^*X^T) = n-q, \tag{3.20}$$

$$X\in P_n, \tag{3.21}$$

then the returned X^* is a q-surface solution of QAP.

Proof. Let X^* be obtained by running Algorithm LagSA. It holds that $X^*\in\Pi_n$ and

$$\text{trace } (\nabla_x H(X^*;\mu_i)(X^*-X^{**})^T) = 0, \tag{3.22}$$

where X^{**} is the optimal solution of the LAP $\min_{X\in\Pi_n}$trace $(\nabla_x H(X^*;\mu_i)X^T)$, or equivalently,

$$\text{trace } (\nabla_x H(X^*;\mu_i)X^{**T}) \le \text{trace } (\nabla_x H(X^*;\mu_i)X^T), \quad \forall X\in\Pi_n. \tag{3.23}$$

Combining the equality (3.22) with the inequalities (3.23), we have

$$\text{trace } (\nabla_x H(X^*;\mu_i)X^{*T}) \le \text{trace } (\nabla_x H(X^*;\mu_i)X^T), \quad \forall X\in\Pi_n. \tag{3.24}$$

Notice that $\nabla_x H(X^*;\mu_i) = 2AX^*B + 2\mu_i X^*$, the inequalities (3.24) can be rewritten as follows:

$$\text{trace } ((AX^*B+\mu_iX^*)X^{*T}) \le \text{trace } ((AX^*B+\mu_iX^*)X^T), \quad \forall X\in\Pi_n. \tag{3.25}$$

According to the fact that trace $(X^*X^{*T}) = n$ and the definition of Π_n^q (3.17), the above inequalities (3.25) imply

$$\text{trace } (AX^*BX^{*T}) \le \text{trace } (AX^*BX^T) - q\mu_i, \quad \forall X\in\Pi_n^q. \tag{3.26}$$

Now we conclude that X^* is a q-surface solution of QAP if it holds that

$$q\mu_i \ge \text{trace } (A(X^*-X)BX^T), \quad \forall X\in\Pi_n^q, \tag{3.27}$$

or equivalently,

$$q\mu_i \ge \max \quad \text{trace } (A(X^*-X)BX^T) \tag{3.28}$$

$$\text{s.t.} \quad X\in\Pi_n^q. \tag{3.29}$$

Due to the high difficulty of solving Problem (3.28)–(3.28), we consider the following convex quadratic programming relaxation:

$$\max \quad (\text{trace } (A(X^*-X)BX^T) + \hat{\lambda}_{\min}\text{trace } (XX^T) - n\hat{\lambda}_{\min}) \tag{3.30}$$

$$\text{s.t.} \quad \text{trace } (X^*X^T) = n-q, \tag{3.31}$$

$$X\in P_n. \tag{3.32}$$

Therefore, as stated in (3.18), a sufficient condition to make the equalities (3.27) true is that $q\mu_i$ is greater than or equal to the optimal value of Problem (3.30)–(3.32). □

Corollary 3.1. *Suppose Algorithm LagSA stops in the i-th outer iteration and returns a solution X^*. If it holds that*

$$\mu_i \ge \max_{q=1,2,\ldots,n}\left(\frac{u_q}{q} - \frac{n}{q}\hat{\lambda}_{\min}\right), \tag{3.33}$$

where $\hat{\lambda}_{\min}$ and u_q are defined in (2.8) and (3.19)–(3.21) *respectively, then the returned X^* is a globally optimal solution of QAP.*

Theorem 3.4. *Suppose Algorithm LagSA stops in the i-th outer iteration and returns a solution X^*. We have*

$$\text{QAP} \le \text{trace } (AX^*BX^{*T}) \le \text{QAP} + \max\left(0, \max_{q=1,2,\ldots,n}\{u_q - n\hat{\lambda}_{\min} - q\mu_i\}\right), \tag{3.34}$$

where QAP *denotes the optimal function value of QAP* (1.1), *$\hat{\lambda}_{\min}$ and u_q are defined in* (2.8) and (3.19)–(3.21), *respectively.*

Proof. Based on (3.26) and the definitions of $\hat{\lambda}_{\min}$ and u_q, for every $X\in\Pi_n^q$, we also have

$$\begin{aligned}
\text{trace } (AX^*BX^{*T}) - \text{trace } (AXBX^T) &\le \text{trace } (A(X^*-X)BX^T) - q\mu_i\\
&= \text{trace } (A(X^*-X)BX^T + \hat{\lambda}_{\min}XX^T)\\
&\quad -n\hat{\lambda}_{\min} - q\mu_i\\
&\le u_q - n\hat{\lambda}_{\min} - q\mu_i,
\end{aligned}$$

where the equality follows from the fact that trace $(XX^T) = n$ for $X\in\Pi_n^q$. Denote by X_q^* the minimizer of Problem (3.16)–(3.17), then

$$\text{trace } (AX^*BX^{*T}) \le \text{trace } (AX_q^*BX_q^{*T}) + u_q - n\hat{\lambda}_{\min} - q\mu_i, \quad \forall q. \tag{3.35}$$

Since $\Pi_n = \bigcup_{q=0,1,\ldots,n}\Pi_n^q$, it follows that

$$\min_{q=0,1,\ldots,n}\text{trace } (AX_q^*BX_q^{*T}) = \text{QAP}. \tag{3.36}$$

Therefore, taking the equality (3.36) into one of the inequalities (3.35), we have

$$\text{trace}\,(AX^*BX^{*T}) \le \text{QAP} + \max_{q=0,1,\ldots,n}\{u_q - n\hat{\lambda}_{\min} - q\mu_i\}.$$

Notice that $u_0 = n\hat{\lambda}_{\min}$. The proof is completed. □

4. Numerical results

We set the parameters $X_0=(1/n)ee^T$, $\mu_0=\max\{-\hat{\lambda}_{\min},0\}$, $\beta=0.2$, $m=4n$, and $\{\mu_i\}$ is updated as follows:

$$\mu_{i+1} = \begin{cases} \mu_i/2 & \text{if } \mu_i \ge \mu_+, \\ 0 & \text{if } 0<\mu_i<\mu_+, \\ \mu_- & \text{if } \mu_i = 0, \\ 2\mu_i & \text{otherwise } \mu_i \le \mu_-, \end{cases} \tag{4.1}$$

where μ_+ and μ_- are constant such that $\min\{-\hat{\lambda}_{\max},0\} \ll \mu_- < 0 < \mu_+ \ll \mu_0$. In this article we set $\mu_+=1$ and $\mu_- = -\mu_k/2$ for some k such that $\mu_+ \le \mu_k = \mu_0/2^k < 2\mu_+$.

To compare with the existing logarithmic smoothing algorithm [12] (LogSA), we first test the same QAP problems, i.e., the Nugent facility location problems [13] and the Steinberg backboard wiring problem [15]. The numerical data were obtained from the QAPLIB [7]. The difficulty is illustrated by the fact that the Nugent problem with 30 variables was solved to optimality only in 2000 [2], while the Steinberg problem with 36 variables was exactly solved in 2001 [4].

Table 1 shows the platform and software that are used by [12] and us, respectively. Table 2 gives the solution values and the computation time solved by LogSA and LagSA. Our results are dominant except for the problem Nugent-20. Especially, we solve the problem Steinberg-36b to optimality. Besides, our computation time is much less.

We report more numerical experiment results on problems from QAPLIB [7], see Table 3 for details, where the second column gives the optimal values or best values found in the literature, denoted by QAP; error:= (LagSA − QAP)/QAP × 100%.

Finally, we point out that the initial points can be generated randomly in P_n. For example, generate $R \in \mathbb{R}^{(n-1)\times n}$ with independent uniform random entries $R_{ij} \in [-1,1]$. Let $Y = R-(1/n)Ree^T$. We have $Ye=0$. Define

$$X = \frac{1}{n}ee^T + \frac{1}{n\times \max\{[1,|e^TY|]\}}\begin{bmatrix} Y \\ -e^TY \end{bmatrix}.$$

It is trivial to verify $Xe = X^Te = e$ and $X \ge 0$, i.e., $X \in P_n$.

Table 1
Platform and software used for numerical comparison.

Platform	LogSA	LagSA
Processor	SGI	Pentium 4 (3.2 GHz)
Operating system	IRIX 6.5	Windows XP
Memory size	1.5 GB	512 MB
Software used	GAMS 20.5, Matlab 6.0	Matlab 7.0
Precision	2.22×10^{-16}	2.22×10^{-16}

Table 2
Comparison of algorithms based on objective values and computation time (seconds).

Problem	QAP	LogSA (value)	LagSA (value)	LogSA (time)	LagSA (time)
nug12	578	590	590	55	2
nug15	1150	1160	1152	120	4
nug20	2570	2578	2604	624	16
nug30	6124	6128	6128	7813	68
ste36a	9526	9680	9638	20230	175
ste36b	15852	16492	15852	29410	218

Table 3
More results of LagSA.

Problem	QAP	LagSA (value/error)	LagSA (time)
had12	1652	1668 (0.97%)	4
had14	2724	2728 (0.15%)	6
had16	3720	3726 (0.16%)	8
had18	5358	5360 (0.04%)	12
had20	6922	6980 (0.84%)	20
lipa20b	27076	27076 (0.00%)	5
lipa30b	151426	151426 (0.00%)	20
lipa40b	476581	476581 (0.00%)	61
lipa50b	1210 244	1210 244 (0.00%)	135
lipa60b	2520 135	2520 135 (0.00%)	280
lipa70b	4603 200	4603 200 (0.00%)	564
lipa80b	7763 962	7763 962 (0.00%)	1001
lipa90b	12490 441	12490 441 (0.00%)	1885
scr20	110030	110978 (0.86%)	23
sko42	15812	15878 (0.42%)	223
sko49	23386	23418 (0.14%)	418
sko56	34458	34500 (0.12%)	816
sko64	48498	48558 (0.12%)	1449
sko72	66256	66430 (0.26%)	2586
sko81	90998	91094 (0.11%)	3973
sko90	115534	115716 (0.16%)	7014
sko100a	152002	152202 (0.13%)	10252
tai12a	224416	224416 (0.00%)	6
tho30	149936	151256 (0.88%)	82
tho40	240516	241312 (0.33%)	255
wil50	48816	48888 (0.15%)	559
wil100	273038	273338 (0.11%)	11455

5. Conclusions and further discussion

Continuation methods are new promising approaches to solving quadratic assignment problems (QAP). In [12] the logarithmic smoothing algorithm (LogSA) was applied to solve QAP and limited numerical results were reported. The Lagrangian smoothing algorithm for QAP was firstly devised in [16]. Numerical experiments indicated its high efficiency. In this article, a new version of this algorithm is proposed with practical stopping criteria. We prove that this algorithm finitely terminates at a KKT point of a continuation subproblem. We introduce the definition of q-surface solution, which is a class of local optimal solutions. Then we establish sufficient condition to make the returned solution a q-surface solution or a globally optimal solution. Numerical results are provided to demonstrate the efficiency.

Our algorithm is determinate since it starts from the center of P_n. Actually, if the maximum number of inner iterations is not large enough, the initial point can be chosen randomly, as shown in the above section. Therefore we can run the algorithm many times from different initial points to achieve high-quality solutions. We notice that if the maximum number of inner iterations is large enough, our algorithm is independent of the initial points since the first subproblem is convex.

Acknowledgements

The author is very grateful to the anonymous referees for many useful suggestions that improve this paper.

References

[1] Anstreicher KM. Recent advances in the solution of quadratic assignment problems. Mathematical Programming, Series B 2003;97:24–42.

[2] Anstreicher KM, Brixius NW, Linderoth J, Goux JP. Solving large quadratic assignment problems on computational grids. Mathematical Programming, Series B 2002;91:563–88.

[3] Bertsekas DP. Nonlinear programming. Belmont, MA: Athena scientific; 1995.

[4] Brixius NW, Anstreicher KM. The Steinberg wiring problem. In: Grötschel M, editor. The Sharpest Cut, Fest-Schrift in Honor of Manfred Padberg's 60th Birthday. SIAM; 2003. p. 331–48.

[5] Frank M, Wolfe P. An algorithm for quadratic programming. Naval Research Logistics Quarterly 1956;3:95–110.

[7] Burkard RE, Karisch SE, Rendl F. QAPLAB—a quadratic assignment problem library. Journal of Global Optimization 1997;10:391–403 (See also 〈www.opt.math.tu-graz.ac.at/~qaplib〉).

[8] Burkard RE, Bönniger T. A heuristic for quadratic Boolean programs with applications to quadratic assignment problems. European Journal of Operational Research 1983;13:374–86.

[9] Çela E. The quadratic assignment problem: theory and algorithms. Dordrecht: Kluwer Academic Publishers; 1998.

[10] Hadley SW, Rendl F, Wolkowicz H. A new lower bound via projection for the quadratic assignment problem. Mathematics of Operations Research 1992;17:727–39.

[11] Loiola EM, Abreu NMM, Boaventura-Netto PO, Hahn P, Querido T. An analytical survey for the quadratic assignment problem. European Journal of Operational Research 2007;176:657–90.

[12] Ng KM. A continuation approach for solving nonlinear optimization problems with discrete variables. PhD thesis, Stanford University; 2002.

[13] Nugent CE, Vollman TE, Ruml J. An experimental comparison of techniques for the assignment of facilities to locations. Operations Research 1968;16:150–73.

[14] Pardalos PM, Rendl F, Wolkowicz H. The quadratic assignment problem: a survey and recent developments. In: Pardalos PM, Wolkowicz H, editors, Quadratic Assignment and Related Problems. DIMACS series in discrete mathematics and theoretical computer science, vol. 16. Rhode Island: AMS; 1994. p. 1–42.

[15] Steinberg L. The backboard wiring problem: a placement algorithm. SIAM Review 1961;3:37–50.

[16] Xia Y. A new continuation approach to quadratic assignment and related problems. In: Yuan Y-X, et al., editors, Proceedings of the eighth national conference of operations research society of China. Hong Kong: Global-Link Informatics Limited; 2006. p. 262–9.