MATHEMATICAL ENGINEERING TECHNICAL REPORTS

Approximation Algorithm for Multidimensional Assignment Problem Minimizing the Sum of Squared Errors

Yusuke KUROKI and Tomomi MATSUI

(Communicated by Kokichi SUGIHARA)

METR 2007–03 January 2007

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.i.u-tokyo.ac.jp/edu/course/mi/index_e.shtml

Approximation Algorithm for Multidimensional Assignment Problem Minimizing the Sum of Squared Errors

Yusuke KUROKI

Graduate School of Information Science and Technology,
The University of Tokyo, Hongo, Bunkyo-ku, Tokyo 113-8656, Japan
kuroky@misojiro.t.u-tokyo.ac.jp

Tomomi MATSUI*

Department of Information and System Engineering,
Faculty of Science and Engineering, Chuo University,
Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan
matsui@ise.chuo-u.ac.jp

Abstract

Given a complete k-partite graph $G = (V_1, V_2, \ldots, V_k; E)$ satisfying $|V_1| = |V_2| = \cdots = |V_k| = n$ and weights of all k-cliques of G, the k-dimensional assignment problem finds a partition of vertices of G into a set of (pairwise disjoint) n k-cliques that minimizes the sum total of weights of the chosen cliques. In this paper, we consider a case in which the weight of a clique is defined by the sum of given weights of edges induced by the clique. Additionally, we assume that vertices of G are embedded in the d-dimensional space $\mathbb{Q}^d$ and a weight of an edge is defined by the square of the Euclidean distance between its two endpoints. We describe that these problem instances arise from a multidimensional Gaussian model of a data association problem.

We show the NP-hardness of the problem when $k = 3$ and $d \geq 2$. Futhermore, we propose a second-order cone programming relaxation of the problem and a polynomial time randomized rounding procedure. We show that the expected objective value obtained by our algorithm is bounded by $(5/2 - 3/k)$ times the optimal value. Our result improves the previously known bound $(4 - 6/k)$ of the approximation ratio.

Key words: multidimensional assignment problem, approximation algorithm, second-order cone programming, data association problem, data fusion, statistical matching

*Supported by Superrobust Computation Project of the 21st Century COE Program "Information Science and Technology Strategic Core."

1. Introduction

Let $\mathcal{F} = \{V_1, V_2, \ldots, V_k\}$ be a family of vertex sets satisfying $|V_1| = |V_2| = \cdots = |V_k| = n$. A complete k-partite graph $G = (V_1, V_2, \ldots, V_k; E)$ is defined by vertex sets $V_1, V_2, \ldots, V_k$, and an edge set $E = \bigcup_{\{U,V\} \in \binom{\mathcal{F}}{2}} \{\{u, v\} \mid u \in U,\ v \in V\}$. A vertex subset Q is called a *clique* (*q-clique*) of G if and only if the complete graph induced by Q is a subgraph of G (and $q = |Q|$). Given weights of all k-cliques of G, the k-dimensional assignment problem finds a partition of vertices of G into a set of (pairwise disjoint) n k-cliques that minimizes the sum total of weights of the chosen n k-cliques.

We introduce the following definitions and assumptions. For any clique Q of G, every edge connecting two vertices in Q is called a *clique edge* of Q. Given an edge weight vector $\boldsymbol{w} \in \mathbb{R}^E$, we define the weight of a clique Q by the sum of weights of clique edges of Q. Additionally, we assume that vertices of G are embedded in the d-dimensional space and that the weight of an edge is defined by the squares of the Euclidean distance between its two endpoints. For the remainder of this paper, we assume that the input of the problem is k n-sets $V_1, V_2, \ldots, V_k$ of rational d-dimensional vectors (i.e., $V_1, V_2, \ldots, V_k \subseteq \mathbb{Q}^d$ and $|V_1| = |V_2| = \cdots = |V_k| = n$). Under that assumption, our problem finds n clusters that minimize the sum of squared errors (the sum of squares of all the distances between points in the same cluster), subject to the constraint that each cluster meets every $V_i \in \mathcal{F}$ in exactly one vertex. The n-clustering problem minimizing the sum of squared errors is discussed in many papers (e.g., [4, 7]). The multidimensional assignment problem for minimizing the sum of squared errors arises from a multidimensional Gaussian model of a data association problem described in Appendix A.

In this paper, we show the NP-hardness of the problem when $k = 3$ and $d \geq 2$ (see Appendix B). Furthermore, we propose a second-order cone programming relaxation of the problem and a polynomial time randomized rounding procedure. We show that the expected objective value obtained by our algorithm is bounded by $(5/2 - 3/k)$ times the optimal value. Our result improves the previously known bound $(4 - 6/k)$ of the approximation ratio obtained by Bandelt, Crama and Spieksma in [2].

When $k = 2$, the k-dimensional assignment problem is a well-known assignment problem and is solvable using the Hungarian method. The 3-dimensional assignment problem has been actively investigated. When weights of all 3-cliques are arbitrary, the problem is a generalization of 3-dimensional matching (3DM) and is therefore NP-hard [12]. The NP-hardness of some subclasses has been addressed in the literature [5, 9, 23]. When edge weights satisfy triangle inequalities, Crama and Spieksma [9] showed that a simple heuristic gives a $(4/3)$-approximation algorithm. For values $k \geq 4$, the k-dimensional assignment problem has been less studied. Early mention of the problem can be found in Haley [13] and Pierskalla [17]. Bandelt, Crama and Spieksma [2] considered cases in which the weights of cliques are not arbitrary, but are instead given as a function of edge weights. When edge weights satisfy triangle inequalities and the weight of a clique is defined by the sum of weights of edges induced by the clique, they showed that there exists a $(2-2/k)$-approximation algorithm. We briefly describe (a modified version of) their algorithm and its approximation ratio in Section 2.1. For more detailed references, see recent survey papers [6] by Burkard and Çela and [24] by Spieksma.

Multidimensional assignment problems arise from many application areas. Pierskalla [16, 17] described some application settings: capital investment, dynamic facility location, and satellite launching. Other applications are enumerated in Frieze and Yadegar [11] and Crama et al. [8]. Recently, multidimensional assignment problems have been revealed for applications as techniques to solve data association problems. For example, in multitarget multisensor surveillance systems, we must associate reports from multisensors to enhance target identification and state estimation. General classes of these problems can be formulated as multidimensional assignment problems [18, 19]. Another example is the integration of market databases. When there is no single source database available for all the information of interest, techniques of integrating different databases are often applied. By integrating multiple source market survey data, the obtained single data-set will have answers to all questions in original surveys. One class of integration methods is known as that of data-fusion procedures or statistical matching [20]. In [22], Soong and de Montigny examined a problem instance of fusing three databases.

2. Formulations and Relaxations

In this section, we formulate the multidimensional assignment problem as an integer linear programming problem and (integer) quadratic programming problems. Finally, we combine our formulations and give a second-order cone programming relaxation.

For the remainder of this paper, we denote the vertex set $V_1 \cup V_2 \cup \cdots \cup V_k$ by $\widehat{V}$. For any vertex subset $V \subseteq \widehat{V}$, $\delta(V)$ denotes the set of edges in E between V and $\widehat{V} \setminus V$. For any disjoint pair of vertex subsets $U, V \subseteq \widehat{V}$, we denote the edge subset $\delta(U) \cap \delta(V)$ by $E(U, V)$ and/or $E(V, U)$. We denote a singleton $\{v\}$ by v for simplicity, when no ambiguity exists. A sequence (e_1, e_2, e_3) of edges of G is called a *triangle* of G if the graph induced by edges $\{e_1, e_2, e_3\}$ is a 3-cycle in G. For any vector $\boldsymbol{x} \in \mathbb{R}^E$ and an edge $\{u, v\} \in E$, we denote the element $x(\{u, v\})$ by $x(u, v)$ and/or $x(v, u)$ for short.

2.1. Integer Linear Programming

We introduce a 0-1 valued variable vector $\boldsymbol{x} \in \{0, 1\}^E$. For an arbitrary edge weight vector $\boldsymbol{w} \in \mathbb{R}^E$, we can formulate the multidimensional assignment problem as

$$\begin{aligned} \text{ILP: min. } & \textstyle\sum_{e \in E} w(e)x(e) \\ \text{s. t. } & \textstyle\sum_{u \in U} x(u, v) = 1 \qquad (\forall U \in \mathcal{F},\ \forall v \in \widehat{V} \setminus U), \qquad (1) \end{aligned}$$

$$
\begin{aligned}
& x(e_1) \geq x(e_2) + x(e_3) - 1 && \text{(for each triangle } (e_1, e_2, e_3) \text{ of } G\text{)}, && (2)\\
& x(e) \in \{0, 1\} && (\forall e \in E).
\end{aligned}
$$

We next demonstrate the correctness of the above formulation. For any $\boldsymbol{x} \in \{0,1\}^E$, we define an edge subset $E(\boldsymbol{x})$ by $\{e \in E \mid x(e) = 1\}$. Let $\boldsymbol{x}$ be a feasible solution of ILP. Then, for any pair $\{U, V\} \in \binom{\mathcal{F}}{2}$, constraints (1) imply that the edge subset $E(\boldsymbol{x}) \cap E(U, V)$ is a perfect matching of the bipartite graph $(U, V; E(U, V))$. Constraints (2) mean that if $[x(e_2) = 1$ and $x(e_3) = 1]$, then $x(e_1) = 1$. Thus constraints (1) and (2) yield that each connected component of $(\widehat{V}, E(\boldsymbol{x}))$ contains k-clique. Because $E(\boldsymbol{x})$ contains $n(1/2)k(k-1)$ edges, the subgraph $(\widehat{V}, E(\boldsymbol{x}))$ consists of pairwise disjoint n k-cliques. The inverse implication is clear.

When we drop constraints (2), we can decompose the obtained problem, denoted by LP, into $(1/2)k(k-1)$ subproblems, each of which is a classical assignment problem defined on a bipartite graph $(U, V; E(U, V))$ for a pair $\{U, V\} \in \binom{\mathcal{F}}{2}$. Consequently, we can solve LP by applying the Hungarian method to each subproblem. In the following, we briefly describe a randomized version of *multiple-hub heuristic* proposed by Bandelt, Crama and Spieksma in [2]. First, we solve the relaxation problem LP and obtain a 0-1 valued optimal solution $\boldsymbol{x}^{\mathrm{LP}}$. Next, we choose a vertex subset $U \in \mathcal{F}$ randomly. Lastly, we construct a graph $G' = (\widehat{V}, \delta(U) \cap E(\boldsymbol{x}^{\mathrm{LP}}))$ and output a family of vertex subsets $\{Q_1, Q_2, \ldots, Q_n\}$ of connected components in G'. Each connected component in G' is a complete bipartite graph $\mathrm{K}_{1,k-1}$ and meets every $V \in \mathcal{F}$ in exactly one vertex. Therefore, the obtained vertex subsets $Q_1, Q_2, \ldots, Q_n$ are pairwise disjoint k-cliques of G. The obtained solution corresponds to a feasible solution $\boldsymbol{X}^{\mathrm{LPR}}$ of ILP defined by

$$
X^{\mathrm{LPR}}(e) = \begin{cases} x^{\mathrm{LP}}(e) & (\forall e \in \delta(U)),\\ \sum_{u \in U} x^{\mathrm{LP}}(u, v) x^{\mathrm{LP}}(u, v') & (\forall e = \{v, v'\} \in E \setminus \delta(U)). \end{cases}
$$

Results of Bandelt, Crama and Spieksma [2] imply the following. Under the assumptions that (i) edge weights are non-negative and

$$
\text{(ii) } \exists \tau \geq 1/2, \text{ for each triangle } (e_1, e_2, e_3) \text{ of } G,\ w(e_1) + w(e_2) \geq (1/\tau) w(e_3), \quad (3)
$$

the expectation of the objective function value of X^{LPR} satisfies that

$$
\mathrm{E}\left[\sum_{e \in E} w(e) X^{\mathrm{LPR}}(e)\right] \leq (2/k)((k-2)\tau + 1) z^*(\mathrm{ILP})
$$

where $z^*(\mathrm{ILP})$ is the optimal value of ILP. We deal with the case in which the weight of an edge is defined by the square of the Euclidean distance between its two endpoints. Therefore, property (3) is satisfied by setting $\tau = 2$. Thus, the approximation ratio of the above algorithm is bounded by $(4 - 6/k)$ for our case.

2.2. Non-Convex Quadratic Programming

We transform the problem ILP to a non-convex quadratic programming problem. In this subsection, we fix a vertex subset $U \in \mathcal{F}$.

Every feasible solution $\boldsymbol{x} \in \{0,1\}^E$ of ILP satisfies that the graph $G' = (\widehat{V}, \delta(U) \cap E(\boldsymbol{x}))$ has n connected components and each component meets every vertex subset in $\mathcal{F}$ in exactly one vertex. Consequently, the variables indexed by edges $E \setminus \delta(U)$ satisfy that

$$\forall e = \{v, v'\} \in E \setminus \delta(U), \;\; x(v, v') = \textstyle\sum_{u \in U} x(u, v) x(u, v'). \tag{4}$$

Using the above equalities, we eliminate variables indexed by $E \setminus \delta(U)$ from the objective function of ILP and obtain the following function:

$$\begin{aligned} &\textstyle\sum_{e \in E} w(e)x(e) = \sum_{e \in \delta(U)} w(e)x(e) + \sum_{e \in E \setminus \delta(U)} w(e)x(e) \\ &= \textstyle\sum_{e \in \delta(U)} w(e)x(e) + \sum_{\{v,v'\} \in E \setminus \delta(U)} \left(w(v,v') \sum_{u \in U} x(u,v)x(u,v')\right). \end{aligned}$$

Because the remaining variables are indexed by $\delta(U)$ and the graph $(\widehat{V}, \delta(U))$ does not include any 3-cycle, we require no constraints (2). By substituting non-negativity constraints for 0-1 constraints, we obtain the following problem:

$$\begin{aligned} \text{NQP}(U)\text{: min. } & \sum_{e \in \delta(U)} w(e)x(e) + \sum_{\{v,v'\} \in E \setminus \delta(U)} \left(w(v,v') \sum_{u \in U} x(u,v)x(u,v')\right) \\ \text{s. t. } & \textstyle\sum_{u \in U} x(u,v) = 1 \quad (\forall v \in \widehat{V} \setminus U), \\ & \textstyle\sum_{v \in V} x(u,v) = 1 \quad (\forall u \in U,\ \forall V \in \mathcal{F} \setminus \{U\}), \\ & x(e) \geq 0 \qquad\qquad\; (\forall e \in \delta(U)). \end{aligned}$$

We show that NQP(U) has a 0-1 valued optimal solution by employing the following randomized rounding procedure, which serves an important role in a later section. Let $\boldsymbol{x} \in \mathbb{R}^{\delta(U)}$ be a feasible solution of NQP(U). For each vertex subset $V \in \mathcal{F} \setminus \{U\}$, the subvector $\boldsymbol{x}|_{E(U,V)}$ of $\boldsymbol{x}$ indexed by $E(U,V)$ is contained in the set

$$\left\{ \widetilde{\boldsymbol{x}} \in \mathbb{R}_+^{E(U,V)} \;\middle|\; \textstyle\sum_{u \in U} \widetilde{x}(u,v) = 1 \ (\forall v \in V),\ \sum_{v \in V} \widetilde{x}(u,v) = 1 \ (\forall u \in U) \right\}.$$

Thus, the subvector $\boldsymbol{x}|_{E(U,V)}$ is contained in the assignment polytope defined on the complete bipartite graph $(U, V; E(U,V))$. Birkhoff–von Neumann's theorem [3, 15] and/or integrality of assignment polytopes yield that we can represent the subvector $\boldsymbol{x}|_{E(U,V)}$ by a convex combination of characteristic vectors of perfect matchings in the bipartite graph $(U, V; E(U,V))$. We then obtain the following randomized rounding procedure.

Procedure 1.

Input: A feasible solution $\boldsymbol{x} \in \mathbb{R}^{\delta(U)}$ of NQP(U).
Output: A 0-1 valued feasible solution $\boldsymbol{X}$ of NQP(U).
For each vertex subset $V \in \mathcal{F} \setminus \{U\}$, execute the following.

Step 1: Represent the subvector $\boldsymbol{x}|_{E(U,V)}$ by a convex combination of characteristic vectors of perfect matchings of the bipartite graph $(U, V; E(U, V))$. We denote the coefficient of convex combination with respect to a perfect matching M by $\lambda(M)$.

Step 2: Choose a perfect matching of $(U, V; E(U, V))$ under the probability function that a perfect matching M is chosen with probability $\lambda(M)$.

Step 3: Set the subvector $\boldsymbol{X}|_{E(U,V)}$ as the characteristic vector of the chosen perfect matching.

Next assume that we applied Procedure 1 to an optimal solution $\boldsymbol{x}^*$ of NQP(U) and obtained a 0-1 valued feasible solution $\boldsymbol{X}$ of NQP(U). Then the expectation of the corresponding objective function value satisfies that

$$\begin{aligned}
&\mathrm{E}\left[\sum_{e\in\delta(V)} w(e)X(e) + \sum_{\{v,v'\}\in E\setminus\delta(U)}\left(w(v,v')\sum_{u\in U} X(u,v)X(u,v')\right)\right]\\
&= \sum_{e\in\delta(U)} w(e)\mathrm{E}[X(e)] + \sum_{\{v,v'\}\in E\setminus\delta(U)}\left(w(v,v')\sum_{u\in U}\mathrm{E}[X(u,v)]\mathrm{E}[X(u,v')]\right)\\
&= \sum_{e\in\delta(U)} w(e)x^*(e) + \sum_{\substack{\{v,v'\}\\ \in E\setminus\delta(U)}}\left(w(v,v')\sum_{u\in U} x^*(u,v)x^*(u,v')\right) = z^*(\mathrm{NQP}(U))
\end{aligned}$$

where $z^*(\mathrm{NQP}(U))$ is the optimal value of NQP(U). The first equality is obtained from the property that every cross term $X(e_1)X(e_2)$ appearing in the objective function satisfies that the pair of random variables $X(e_1)$ and $X(e_2)$ is independent. Because NQP(U) has a 0-1 valued feasible solution whose objective value is less than or equal to the above expectation, the above equalities imply that NQP(U) has a 0-1 valued optimal solution.

For any feasible solution $\boldsymbol{x} \in \{0,1\}^E$ of ILP, the subvector $\boldsymbol{x}|_{\delta(U)}$ is feasible to NQP(U). Conversely, if we have a 0-1 valued feasible solution $\boldsymbol{x} \in \{0,1\}^{\delta(U)}$ of NQP(U), we can construct a feasible solution of ILP using equalities (4). These transformations give a bijection between the feasible set of ILP and the set of 0-1 valued feasible solutions of NQP(U). The objective function values of corresponding pair of solutions are equivalent. Therefore, we can construct an optimal solution of ILP from a 0-1 valued optimal solution of NQP(U) using equalities (4).

2.3. Integer Quadratic Programming

In this subsection, we reformulate NQP(U) as an integer programming problem with a convex quadratic objective function. We also fix a subset $U \in \mathcal{F}$ throughout this subsection. In the remainder of this section, we use the assumption that vertices in $\widehat{V}$ are embedded in $\mathbb{Q}^d$ and the weight of an edge is defined by the square of the Euclidean distance between its two endpoints. For any vertex $v \in \widehat{V}$, we denote the position (in $\mathbb{Q}^d$) of v by $\boldsymbol{v} \in \mathbb{Q}^d$. For any clique Q of G, we denote the weight of Q by $w(Q)$.

Let $\boldsymbol{x} \in \{0,1\}^{\delta(U)}$ be a 0-1 valued feasible solution of NQP(U) and $\mathcal{Q} = \{Q_1, Q_2, \ldots, Q_n\}$ be the set of corresponding n k-cliques. For any vertex $u \in U$, $Q(u)$ denotes a unique clique in $\mathcal{Q}$ including u. The objective function value of NQP(U) with respect to $\boldsymbol{x}$ is the sum total of clique weights and is therefore equal to

$$\textstyle\sum_{u \in U} w(Q(u)) = \sum_{e \in \delta(U)} w(e)x(e) + \sum_{u \in U} w(Q(u) \setminus u).$$

For any vertex $u \in U$, the clique $Q(u)$ meets every subset $V \in \mathcal{F} \setminus \{U\}$ in exactly one vertex in the singleton $Q(u) \cap V$, whose position (in $\mathbb{Q}^d$) is denoted by $\sum_{v \in V} x(u,v)\boldsymbol{v}$, because the equality $\sum_{v \in V} x(u,v) = 1$ holds. For any pair $\{V, V'\} \in \binom{\mathcal{F} \setminus \{U\}}{2}$, the clique $Q(u)$ has a unique clique edge in $E(V, V')$ connecting vertices in $Q(u) \cap V$ and $Q(u) \cap V'$. Therefore, the weight of the edge is equal to

$$\left\| \textstyle\sum_{v \in V} x(u,v)\boldsymbol{v} - \sum_{v' \in V'} x(u,v')\boldsymbol{v}' \right\|^2.$$

From the above, the sum total of clique weights, $\sum_{u \in U} w(Q(u))$, is given as

$$\sum_{e \in \delta(U)} w(e)x(e) + \sum_{u \in U} \sum_{\{V,V'\} \in \binom{\mathcal{F} \setminus \{U\}}{2}} \left\| \sum_{v \in V} x(u,v)\boldsymbol{v} - \sum_{v' \in V'} x(u,v')\boldsymbol{v}' \right\|^2.$$

Employing the above function, we obtain the following integer quadratic programming formulation of our problem:

$$\begin{array}{ll} \text{IQP}(U): & \\ \text{min.} \displaystyle\sum_{e \in \delta(U)} w(e)x(e) + \sum_{u \in U} \sum_{\{V,V'\} \in \binom{\mathcal{F} \setminus \{U\}}{2}} \left\| \sum_{v \in V} x(u,v)\boldsymbol{v} - \sum_{v' \in V'} x(u,v')\boldsymbol{v}' \right\|^2 & \\ \text{s. t.} \textstyle\sum_{u \in U} x(u,v) = 1 \quad (\forall v \in \widehat{V} \setminus U), & \\ \qquad \textstyle\sum_{v \in V} x(u,v) = 1 \quad (\forall u \in U,\ \forall V \in \mathcal{F} \setminus \{U\}), & \\ \qquad x(e) \in \{0,1\} \qquad (\forall e \in \delta(U)). & \end{array}$$

Different from NQP(U), we cannot drop 0-1 constraints in IQP(U). An advantage of this formulation is that the objective function is a convex quadratic function. For that reason, the continuous relaxation problem, obtained by substituting non-negativity constraints for 0-1 constraints, is a convex quadratic programming problem that is solvable efficiently.

2.4. Second-Order Cone Programming Relaxation

Lastly, we combine formulations ILP and IQP(U) and construct a second-order cone programming (SOCP) relaxation, which, we hope, provides a better lower bound.

Here we note that we do not fix a vertex subset $U \in \mathcal{F}$ in this subsection. By introducing an artificial variable z, our relaxation problem is described as follows:

$$
\begin{array}{ll}
\text{SOCPR:} & \\
\text{min. } z & \\
\text{s. t. } z \geq \sum_{\{u,v\} \in E} \|\boldsymbol{v} - \boldsymbol{u}\|^2 x(u,v), & \\
\quad z \geq \sum_{\{u,v\} \in \delta(U)} \|\boldsymbol{v} - \boldsymbol{u}\|^2 x(u,v) & \\
\quad\quad + \sum_{u \in U} \sum_{\{V,V'\} \in \binom{\mathcal{F} \setminus \{U\}}{2}} \left\| \sum_{v \in V} x(u,v)\boldsymbol{v} - \sum_{v' \in V'} x(u,v')\boldsymbol{v}' \right\|^2 & (\forall U \in \mathcal{F}), \\
\quad \sum_{u \in U} x(u,v) = 1 \quad (\forall U \in \mathcal{F},\ \forall v \in \widehat{V} \setminus U), & \\
\quad x(e) \geq 0 \quad (\forall e \in E). &
\end{array}
$$

As is well-known, the above problem can be transformed to a second-order cone programming problem, which is solvable within any given gap ϵ in polynomial time using an interior point method (see a recent survey paper [1]).

3. Randomized Approximation Algorithm

In this section, we propose a randomized approximation algorithm.

Algorithm 2.

Input: Subsets $V_1, V_2, \ldots, V_k \subseteq \mathbb{Q}^d$ satisfying $|V_1| = |V_2| = \cdots = |V_k| = n$.
Output: A feasible solution $\boldsymbol{X}$ of ILP.
Step 1: Solve SOCPR and obtain an optimal solution $(z^*, \boldsymbol{x}^*)$.
Step 2: Randomly choose a vertex set $U \in \mathcal{F}$.
Step 3: Apply Procedure 1 to the subvector $\boldsymbol{x}^*|_{\delta(U)}$ and obtain 0-1 valued vector $\boldsymbol{X}_U$ indexed by $\delta(U)$.
Step 4: Output a 0-1 valued vector $\boldsymbol{X} \in \{0,1\}^E$ defined as

$$
X(e) = \begin{cases} X_U(e) & (\forall e \in \delta(U)), \\ \sum_{u \in U} X_U(u,v) X_U(u,v') & (\forall e = \{v,v'\} \in E \setminus \delta(U)). \end{cases}
$$

For executing Procedure 1 in Step 3, we need to represent the subvector $\boldsymbol{x}^*|_{E(U,V)}$ by a convex combination of characteristic vectors of perfect matchings in the bipartite graph $(U, V; E(U,V))$ for each $V \in \mathcal{F} \setminus \{U\}$. We can find coefficients for convex combination by applying (an unweighted version of) the Hungarian method $\mathrm{O}(n^2)$ times. Therefore, employing an $\mathrm{O}(n^{2.5})$ algorithm for the assignment problem in [14], Step 3 requires $\mathrm{O}(kn^{4.5})$ computational time. Appendix C gives an $\mathrm{O}(kn^4)$ time algorithm for Step 3. Although Step 4 requires $\mathrm{O}(k^2n^3)$ time, we need not execute Step 4 to output n k-cliques (vertex subsets of G), which requires only $\mathrm{O}(kn)$ time.

The following theorem is our main result.

Theorem 3. Algorithm 2 *finds a feasible solution of ILP such that the expectation of the corresponding objective function value is less than or equal to* $(5/2 - 3/k)z^{**}$ *where* z^{**} *is the optimal value of the multidimensional assignment problem defined by subsets* $V_1, \dots, V_k \subseteq \mathbb{Q}^d$ *satisfying* $|V_1| = \cdots = |V_k| = n$.

Proof. Let $(z^*, \boldsymbol{x}^*)$ be an optimal solution of SOCPR, and $\boldsymbol{X}$ be a solution obtained by Algorithm 2. The feasibility of $\boldsymbol{X}$ is clear. The expectation of the corresponding objective function value satisfies that

$$
\begin{aligned}
&\mathrm{E}\left[\textstyle\sum_{e \in E} w(e)X(e)\right] \\
&= \frac{1}{k} \sum_{U \in \mathcal{F}} \mathrm{E}\left[\sum_{e \in \delta(U)} w(e) X_U(e) + \sum_{\substack{\{v,v'\} \\ \in E \setminus \delta(U)}} \left(w(v,v') \sum_{u \in U} X_U(u,v) X_U(u,v') \right) \right] \\
&= \frac{1}{k} \sum_{U \in \mathcal{F}} \left(\sum_{e \in \delta(U)} w(e) \mathrm{E}[X_U(e)] + \sum_{\substack{\{v,v'\} \\ \in E \setminus \delta(U)}} \left(w(v,v') \sum_{u \in U} \mathrm{E}[X_U(u,v)] \mathrm{E}[X_U(u,v')] \right) \right) \\
&= \frac{1}{k} \sum_{U \in \mathcal{F}} \left(\sum_{e \in \delta(U)} w(e) x^*(e) + \sum_{\{v,v'\} \in E \setminus \delta(U)} \left(w(v,v') \sum_{u \in U} x^*(u,v) x^*(u,v') \right) \right) \\
&= \frac{1}{k} \sum_{U \in \mathcal{F}} \sum_{e \in \delta(U)} w(e) x^*(e) \\
&+ \frac{1}{k} \sum_{U \in \mathcal{F}} \sum_{\{V,V'\} \in \binom{\mathcal{F} \setminus \{U\}}{2}} \sum_{v \in V} \sum_{v' \in V'} \sum_{u \in U} w(v,v') x^*(u,v) x^*(u,v').
\end{aligned}
$$

The assumption related to edge weights implies that

$\forall U \in \mathcal{F}, \forall u \in U, \forall \{v, v'\} \in E \setminus \delta(U),$

$$
\begin{aligned}
w(v,v') &= \|\boldsymbol{v}' - \boldsymbol{v}\|^2 = \|(\boldsymbol{u} - \boldsymbol{v}) - (\boldsymbol{u} - \boldsymbol{v}')\|^2 \\
&= \|\boldsymbol{u} - \boldsymbol{v}\|^2 + \|\boldsymbol{u} - \boldsymbol{v}'\|^2 - 2(\boldsymbol{u} - \boldsymbol{v})^\top (\boldsymbol{u} - \boldsymbol{v}') \\
&= (w(v,u) + w(v',u)) - 2(\boldsymbol{u} - \boldsymbol{v})^\top (\boldsymbol{u} - \boldsymbol{v}'). \qquad (5)
\end{aligned}
$$

For dealing with the first term of (5), we use the equalities

$$
\begin{aligned}
&\sum_{v\in V}\sum_{v'\in V'}(w(v,u)+w(v',u))x^*(v,u)x^*(v',u)\\
&=\sum_{v\in V}\sum_{v'\in V'}w(v,u)x^*(v,u)x^*(v',u)+\sum_{v\in V}\sum_{v'\in V'}w(v',u)x^*(v,u)x^*(v',u)\\
&=\sum_{v\in V}\left(w(v,u)x^*(v,u)\sum_{v'\in V'}x^*(v',u)\right)+\sum_{v'\in V'}\left(w(v',u)x^*(v',u)\sum_{v\in V}x^*(v,u)\right)\\
&=\sum_{v\in V}w(v,u)x^*(v,u)+\sum_{v'\in V'}w(v',u)x^*(v',u)
\end{aligned}
$$

and obtain that

$$
\begin{aligned}
&\sum_{u\in U}\sum_{\{V,V'\}\in\binom{\mathcal{F}\setminus\{U\}}{2}}\sum_{v\in V}\sum_{v'\in V'}(w(v,u)+w(v',u))x^*(v,u)x^*(v',u)\\
&=\sum_{u\in U}\sum_{\{V,V'\}\in\binom{\mathcal{F}\setminus\{U\}}{2}}\left(\sum_{v\in V}w(v,u)x^*(v,u)+\sum_{v'\in V'}w(v',u)x^*(v',u)\right)\\
&=(k-2)\sum_{u\in U}\sum_{V\in\mathcal{F}\setminus\{U\}}\sum_{v\in V}w(v,u)x^*(v,u)=(k-2)\sum_{e\in\delta(U)}w(e)x^*(e). \qquad (6)
\end{aligned}
$$

Next, we consider the last term in (5). It is easy to show that

$$
\begin{aligned}
&-2\sum_{v\in V}\sum_{v'\in V'}(\boldsymbol{u}-\boldsymbol{v})^\top(\boldsymbol{u}-\boldsymbol{v}')x^*(v,u)x^*(v',u)\\
&=-2\left(\sum_{v\in V}x^*(v,u)(\boldsymbol{u}-\boldsymbol{v})\right)^\top\left(\sum_{v'\in V'}x^*(v',u)(\boldsymbol{u}-\boldsymbol{v}')\right)\\
&=-2\left(\sum_{v\in V}x^*(v,u)\boldsymbol{u}-\sum_{v\in V}x^*(v,u)\boldsymbol{v}\right)^\top\left(\sum_{v'\in V'}x^*(v',u)\boldsymbol{u}-\sum_{v'\in V'}x^*(v',u)\boldsymbol{v}'\right)\\
&=-2\left(\boldsymbol{u}-\sum_{v\in V}x^*(v,u)\boldsymbol{v}\right)^\top\left(\boldsymbol{u}-\sum_{v'\in V'}x^*(v',u)\boldsymbol{v}'\right)\\
&\le\frac{1}{2}\left\|\left(\boldsymbol{u}-\sum_{v\in V}x^*(v,u)\boldsymbol{v}\right)-\left(\boldsymbol{u}-\sum_{v'\in V'}x^*(v',u)\boldsymbol{v}'\right)\right\|^2\\
&=\frac{1}{2}\left\|\sum_{v\in V}x^*(v,u)\boldsymbol{v}-\sum_{v'\in V'}x^*(v',u)\boldsymbol{v}'\right\|^2, \qquad (7)
\end{aligned}
$$

where the above inequality is obtained from the fact that $\forall\boldsymbol{p},\forall\boldsymbol{q}\in\mathbb{R}^d$, the inequality $-2\boldsymbol{p}^\top\boldsymbol{q}\le(1/2)\|\boldsymbol{p}-\boldsymbol{q}\|^2$ holds. Equality (6) and inequality (7) yield an upper bound of the expectation as follows:

$$
\begin{aligned}
&\mathrm{E}\left[\sum_{e\in E} w(e)X(e)\right] = \frac{1}{k}\sum_{U\in\mathcal{F}}\sum_{e\in\delta(U)} w(e)x^*(e) \\
&+\frac{1}{k}\sum_{U\in\mathcal{F}}\sum_{u\in U}\sum_{\{V,V'\}\in\binom{\mathcal{F}\setminus\{U\}}{2}}\sum_{v\in V}\sum_{v'\in V'} w(v,v')x^*(v,u)x^*(v',u) \\
&\leq \frac{1}{k}\sum_{U\in\mathcal{F}}\sum_{e\in\delta(U)} w(e)x^*(e) + \frac{k-2}{k}\sum_{U\in\mathcal{F}}\sum_{e\in\delta(U)} w(e)x^*(e) \\
&+\frac{1}{2k}\sum_{U\in\mathcal{F}}\sum_{u\in U}\sum_{\{V,V'\}\in\binom{\mathcal{F}\setminus\{U\}}{2}}\left\|\sum_{v\in V} x^*(v,u)\boldsymbol{v} - \sum_{v'\in V'} x^*(v',u)\boldsymbol{v'}\right\|^2 \\
&= \left(\frac{1}{k}+\frac{k-2}{k}-\frac{1}{2k}\right)\sum_{U\in\mathcal{F}}\sum_{e\in\delta(U)} w(e)x^*(e) \\
&+\frac{1}{2k}\sum_{U\in\mathcal{F}}\left(\sum_{e\in\delta(U)} w(e)x^*(e)\right. \\
&\qquad\left.+\sum_{u\in U}\sum_{\{V,V'\}\in\binom{\mathcal{F}\setminus\{U\}}{2}}\left\|\sum_{v\in V} x^*(v,u)\boldsymbol{v} - \sum_{v'\in V'} x^*(v',u)\boldsymbol{v'}\right\|^2\right) \\
&\leq \frac{2k-3}{2k}\,2\sum_{e\in E} w(e)x^*(e) + \frac{1}{2k}\sum_{U\in\mathcal{F}} z^* \leq \frac{2k-3}{k}z^* + \frac{1}{2}z^* \leq \left(\frac{5}{2}-\frac{3}{k}\right) z^{**}.
\end{aligned}
$$

Therefrom, we obtained the desired result. □

Appendix

A. A Multidimensional Gaussian Model of a Data Association Problem

In this section, we show that our model of a multidimensional assignment problem arises from a simple probabilistic framework of the data association problem. Assume that there are n objects (targets, randomly chosen customers, etc.) and k data-sets (observations obtained by radar or global positioning system, results of questionnaires, etc.) such that each data-set consists of n reports (observations) corresponding to n objects. For fusing k data-sets, we must find a partition of all the reports into (pairwise disjoint) n k-sets such that each subset of reports meets every data-set in exactly one report because we do not know the correspondence (matching) between reports for any pair of data-sets. We assume that each report of object i might be independently and identically distributed from d-dimensional normal distribution $\mathrm{N}(\boldsymbol{\theta}_i, \Sigma)$. In the following, we assume that Σ is the d-dimensional identity matrix for simplicity. When we have k reports $Q = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_k\} \subseteq \mathbb{R}^d$ of object i, the maximum likelihood estimator (MLE) of $\boldsymbol{\theta}_i$ is the center of gravity $(1/k) \sum_{\boldsymbol{v} \in Q} \boldsymbol{v}$ because Σ is the identity matrix and the corresponding log-likelihood is

$$-(1/2) \sum_{\boldsymbol{u} \in Q} \|(1/k) \sum_{\boldsymbol{v} \in Q} \boldsymbol{v} - \boldsymbol{u}\|^2 + c = -(1/2k) \sum_{\{\boldsymbol{u},\boldsymbol{v}\} \in \binom{Q}{2}} \|\boldsymbol{v} - \boldsymbol{u}\|^2 + c$$

where $c = -(1/2)kd \log(2\pi)$. Given a partition $\{Q_1, Q_2, \ldots, Q_n\}$ of all the reports such that each set corresponds to reports from a common potential object, the MLE of the set of n parameters are

$$\{(1/k) \sum_{\boldsymbol{v} \in Q_i} \boldsymbol{v} \mid i \in \{1, 2, \ldots, n\}\},$$

and the corresponding conditional log-likelihood is

$$-(1/2k) \sum_{i=1}^{n} \sum_{\{\boldsymbol{u},\boldsymbol{v}\} \in \binom{Q_i}{2}} \|\boldsymbol{v} - \boldsymbol{u}\|^2 + nc. \tag{8}$$

From the above, we can find the MLE of the set of n parameters using the following two steps: first, solve the multidimensional assignment problem and find pairwise disjoint n k-cliques $\{Q_1, Q_2, \ldots, Q_n\}$ which maximizes the log-likelihood (8); second, for each subset in $\{Q_1, Q_2, \ldots, Q_n\}$, output the center of gravity of contained reports. For this model, we must solve a multidimensional assignment problem that minimizes the sum total of weights of clique edges under the assumptions that vertices of G are embedded in the d-dimensional space $\mathbb{R}^d$ and that the weight of an edge is defined by the square of the Euclidean distance between its two endpoints.

B. Hardness Result

In this section, we will prove NP-hardness of the multidimensional assignment problem minimizing the sum of squared errors when $k = 3$ and $d \geq 2$. A similar discussion can be found in Spieksma–Woeginger [23], where it is shown that a geometric three-dimensional assignment problem (given three n-sets ($\subseteq \mathbb{Z}^2$), find an three-dimensional assignment minimizing the total circumference) is NP-hard.

We introduce an NP-complete problem to which we provide a reduction from our problem.

Problem 4 (PLANAR 3DM)**.**

Instance: Three pairwise disjoint sets X, Y, and Z with $|X| = |Y| = |Z| = q$ and a set $T \subseteq X \times Y \times Z$ such that every element of $X \cup Y \cup Z$ occurs in at most three triplets in T and such that a bipartite graph $H = (T, X \cup Y \cup Z; F)$, where an edge connects a triplet and its element, is planar.

Question: Does there exist a subset T' of q triplets in T such that each element of $X \cup Y \cup Z$ is contained in precisely one triplet of T'? □

Example 5. Let us consider the following instance of PLANAR 3DM:
$X = \{x_1, x_2, x_3\}$, $Y = \{y_1, y_2, y_3\}$, $Z = \{z_1, z_2, z_3\}$, and $T = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8\}$ where $t_1 = (x_1, y_1, z_1)$, $t_2 = (x_1, y_1, z_2)$, $t_3 = (x_3, y_1, z_2)$, $t_4 = (x_3, y_3, z_3)$, $t_5 = (x_2, y_2, z_3)$, $t_6 = (x_2, y_3, z_1)$, $t_7 = (x_1, y_2, z_1)$, and $t_8 = (x_3, y_2, z_3)$. For this instance, $T' = \{t_2, t_6, t_8\}$ is a feasible solution. □

Proposition 6 (Dyer–Frieze [10])**.** *Problem* PLANAR 3DM *is NP-complete.* □

Next, we introduce *planar grid embeddings* of planar graphs. Given a planar graph H with the maximum degree at most 4, a planar grid embedding of H maps vertices to mutually distinct integer grid points in $\mathbb{Z}^2$ and edges to non-intersecting integer grid paths. The following proposition says that we can construct a planar grid embedding efficiently.

Proposition 7 (Tamassia–Tollis [25])**.** *Given a planar graph H with h vertices and maximum degree at most 4, a planar grid embedding of H can be computed in $\mathrm{O}(h)$ time and the area of the embedding is $\mathrm{O}(h^2)$.* □

Now, we will show the decision version of our problem is NP-complete.

Problem 8.

Instance: A complete tripartite graph $C = (R, G, B; E)$ satisfying $|R| = |G| = |B| = n$ and $R, G, B \subseteq \mathbb{Z}^2$, and a positive integer θ.

Question: Does there exist a partition of vertices of C into a set of (pairwise disjoint) n 3-cliques with total weights at most θ where total weights are defined by the sum of the squared distances between two endpoints of all the edges induced in C by the chosen n 3-cliques? □

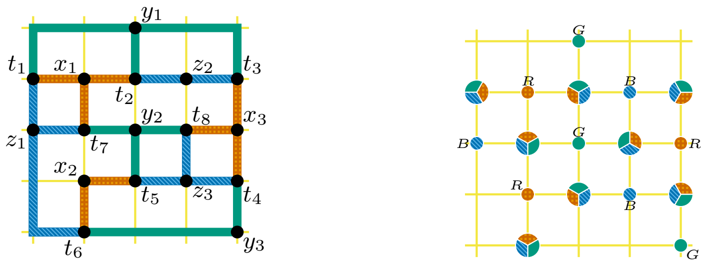
Proposition 9. Problem 8 *is NP-complete.*

Proof. It is easy to see that Problem 8 is in class NP since we can check in polynomial time whether every point appears exactly one 3-clique and whether their weights are at most θ.

We transform PLANAR 3DM to Problem 8. Let X, Y and Z satisfying $|X| = |Y| = |Z| = q$ and $T \subseteq X \times Y \times Z$ constitute any instance of PLANAR 3DM. We must construct point sets R, G, and B with $|R| = |G| = |B| = n$ and positive integer θ such that the complete tripartite graph $C = (R, G, B; E)$ has a partition of weights at most θ if and only if there exists a subset $T' \subseteq T$ of size q uses all points from $X \cup Y \cup Z$ exactly once. For the sake of convenience, we use color red, green, and blue for R, G, and B respectively.

Consider the planar bipartite graph H constructed from the instance of PLANAR 3DM. We denote by *X-path* the bipartite edge of H emanating from X, by *Y-path* the edge from Y, and by *Z-path* the edge from Z.

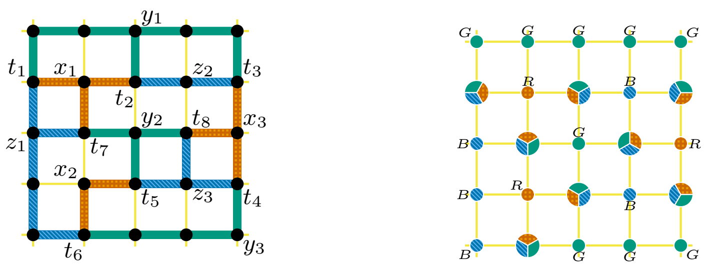
First, we compute a planar grid embedding of H (see Proposition 7); we denote this embedding by H_{E} (e.g. Figure 1a). Let us call the points of H_{E} corresponding to $X \cup Y \cup Z$ *ground points*, and corresponding to T *tricolored points*. Concurrently, we prepare colored vertex set $V_{\mathrm{E}} \subseteq \mathbb{Z}^2$ as follows: place a red vertex (●R) on each ground point of X, green (●G) on each of Y, blue (●B) on each of Z, and place three vertices, each of which is a red, a green, and a blue vertex (●) on every tricolored point together (cf. Figure 1b).



(a) H_{E}. (b) V_{E}.

Figure 1. A planar grid embedding and a colored vertex set of the instance appearing in Example 5 (yellow lines represent integer grids).

Second, we transform H_{E} into H_{S} by subdividing the integer grid paths into integer unit segments (cf. Figure 2a). Let us name the newly inserted points of H_{S} *monocolored points*. We also transform V_{E} into V_{S} by adding some colored vertices as follows: place a red vertex (●R) on every monocolored point along X-path, green (●G) along Y-path, and blue (●B) along Z-path (cf. Figure 2b).



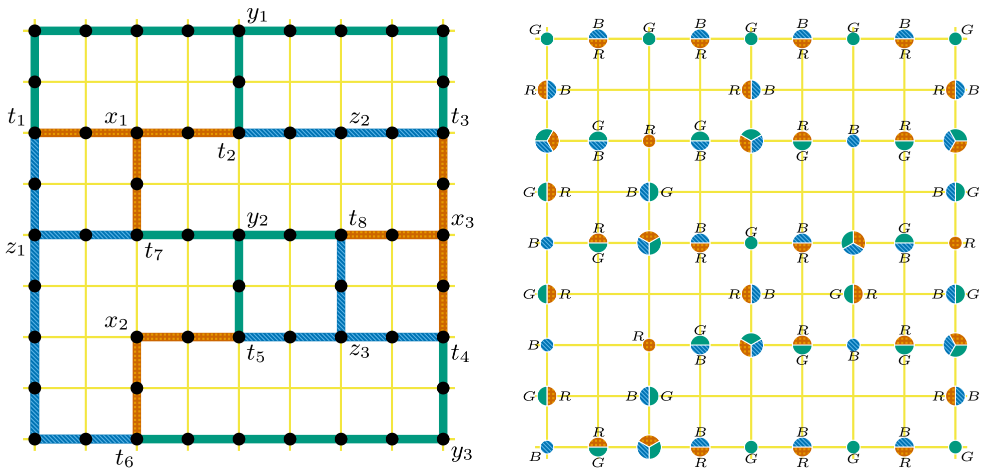
(a) H_{S}. (b) V_{S}.

Figure 2. The subdivision of H_{E} and the corresponding colored vertex set.

Thirdly, we multiply all coordinates of H_{S} and V_{S} by 2. Then, subdivide all segments into integer unit segments again and obtain a new plane graph H_{D} (cf. Figure 3a). We denote the newly inserted points of H_{D} by *bicolored points*. We also construct V_{D} from V_{S} as follows: place a green vertex and a blue vertex together (G●B) on each bicolored point along X-path, blue and red (B●R) along Y-path, and red and green (R●G) along Z-path (cf. Figure 3b).

Finally, span edges appropriately among V_{D} and obtain a complete tripartite graph $C = (R, G, B; E)$.

Due to the way of construction, n is defined as the sum of q and the number of segments in H_{S}. We define θ by $2(n - q)$.



(a) H_{D}. (b) V_{D}.

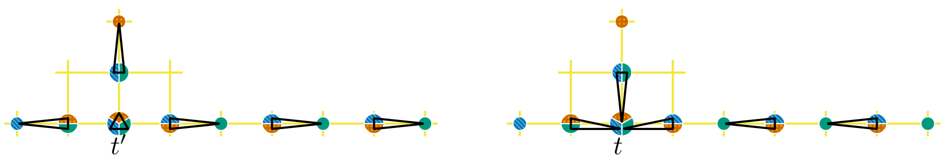
Figure 3. The subdivision of H_{S} and the corresponding colored vertex set.

Now, we must argue that a feasible solution of the PLANAR 3DM corresponds to a feasible solution of Problem 8 whose weight is at most θ and vice versa.

(PLANAR 3DM $\rightarrow$ Problem 8). Let $T' \subseteq T$ be a feasible solution of the PLANAR 3DM.

For each triplet t' in T', we choose a 3-clique with weight 0, i.e., we make a 3-clique using the three colored vertices at the tricolored point t'. Then along the paths emanating from t', from the next point of t' until the element of t', we construct 3-cliques with weights 2 using the next (unused) three colored vertices, one after another (see Figure 4a).

On the other hand, for each t in $T \setminus T'$, we construct three 3-cliques around t using three colored vertices at t separately. Then along paths emanating from t, until the point just in front of the element of t, we construct 3-cliques with weights 2 using the next (unused) three colored vertices, alternately (see Figure 4b).



(a) Around the tricolored point $t' \in T'$. (b) Around the tricolored point $t \in T \backslash T'$.

Figure 4. How to choose 3-cliques (black narrow triangles represent 3-cliques).

Since the set T' is a feasible solution of PLANAR 3DM, every element of $X \cup Y \cup Z$ is contained in precisely one triplet of T'. As a result of the above construction of 3-cliques, only the 3-cliques along the paths emanating from T' can cover the colored vertices on the ground points. Moreover, total weight is equal to the number of segments in H_{D}, i.e., $2(n-q)=\theta$. Therefore, we can construct a positive certificate for Problem 8 (cf. Figure 5).

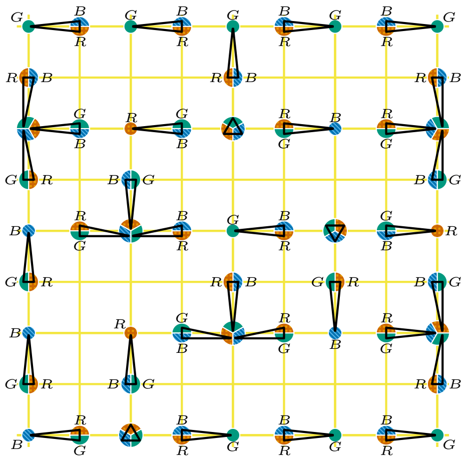


Figure 5. How to construct 3-cliques for Figure 3b.

<u>(PLANAR 3DM $\leftarrow$ Problem 8).</u> Suppose a solution of Problem 8 with weights at most θ exists.

We call edges with weight 0 *internal edges* and call the other edges *external edges*. For example, an edge connecting between two colored vertices on a bicolored point is an internal edge, and another edge adjacent to a colored vertex on a bicolored point is an external edge. For a 3-clique, the induced subgraph of C by the 3-clique is called a *triangle*, and an edge of triangle is called a *triangle edge*.

Now, we would like to show the following statement.

Claim 10. *For any bicolored point, two colored vertices on the point are contained in a unique* 3*-clique. Moreover, exactly one external edge with weight* 1 *leaves from every colored vertex on any bicolored point as a triangle edge.*

For every external edge, weight of the edge is 1 or more. Since there are only two colored vertices on any bicolored point, each of them must adopt at least one external edge as a triangle edge to construct a 3-clique. If we sum up the weights of the external edges, then we obtain at least twice the number of bicolored points. In this calculation, we may worry about counting edges more than once. Especially, we need to consider edges connecting between a colored vertex on a bicolored point and a colored vertex on another bicolored point. Since weights of such edges are at least 2, we use the half of the weights for summation when we sum up the weights of such edges. Because of the way of construction of V_{D}, the number of colored vertices on bicolored points is $2(n-q)$. From the above, we obtain the following inequality:

$$[\text{weight of any feasible solution of Problem 8}] \geq 2(n-q) = \theta.$$

We have the solution of Problem 8 with weights at most θ, thus the weight of this solution is equal to θ.

Therefore, from every colored vertex on any bicolored point, only one external edge emanates as a triangle edge. The other triangle edges must be internal edges. Thus, every two colored vertices on any bicolored point are contained in a unique 3-clique, and thus every edge is counted at most once. We can state Claim 10.

Hence, we have only two choices around bicolored points as shown in Figure 6.

(a) (b)

Figure 6. Only two choices to construct 3-cliques around bicolored points.

Thinking about remaining $3q$ vertices and using the mapping in Figure 4, we obtain a feasible solution for PLANAR 3DM. $\square$

Consequently, we obtain the following theorem.

Theorem 11. *The* 3*-dimensional assignment problem minimizing the squared errors defined by subsets* $V_1, V_2, V_3 \subseteq \mathbb{Q}^d$ *satisfying* $|V_1| = |V_2| = |V_3| = n$ *and* $d \geq 2$ *is NP-hard.*

C. A Decomposition Algorithm

In this section, we describe an efficient algorithm for representing a fractional solution of assignment problem (a doubly stochastic matrix) by a convex combination of integer solutions (permutation matrices), which appears in Step 1 of Procedure 1 and Step 3 of Algorithm 2.

Given two n-sets U and V, $G = (U, V; E)$ denotes a complete bipartite graph where $E := \{\{u,v\} \mid u \in U,\ v \in V\}$. For any vector $\boldsymbol{x} \in \mathbb{R}^E$ and an edge $\{u,v\} \in E$, we denote the element $x(\{u,v\})$ by $x(u,v)$ for short. For any $\boldsymbol{x} \in \mathbb{R}^E_+$, we define an edge subset $E(\boldsymbol{x})$ by $\{e \in E \mid x(e) > 0\}$. Also we denote the characteristic vector of $E' \subseteq E$ as $\boldsymbol{\chi}(E')$, that is, $\chi(E')(e) = 1$ if $e \in E'$ and $\chi(E')(e) = 0$ if $e \notin E'$. The assignment polytope of G is defined as

$$\left\{ \widetilde{\boldsymbol{x}} \in \mathbb{R}^E_+ \mid \textstyle\sum_{u \in U} \widetilde{x}(u,v) = 1 \ (\forall v \in V),\ \textstyle\sum_{v \in V} \widetilde{x}(u,v) = 1 \ (\forall u \in U) \right\}.$$

For any (fractional) vector $\boldsymbol{x}_0$ in the assignment polytope of G, Hall's theorem (e.g. [21: p. 379]) says that the bipartite graph $(U, V; E(\boldsymbol{x}_0))$ has at least one perfect matching. Besides, we can refer to more general statement.

Proposition 12. *For every constant $c > 0$, and for every vector $\boldsymbol{x}$ in the set*

$$\left\{\widetilde{\boldsymbol{x}} \in \mathbb{R}_+^E \mid \sum_{u\in U} \widetilde{x}(u,v) = c \ (\forall v \in V),\ \ \sum_{v\in V} \widetilde{x}(u,v) = c \ (\forall u \in U)\right\},$$

the bipartite graph $(U, V; E(\boldsymbol{x}))$ has at least one perfect matching. $\square$

Not only we can find one perfect matching in the graph $(U, V; E(\boldsymbol{x}_0))$, but also we can represent the vector $\boldsymbol{x}_0$ by a convex combination of characteristic vectors of perfect matchings in $(U, V, E(\boldsymbol{x}_0))$ according to Birkhoff–von Neumann's theorem [3, 15] and/or integrality of assignment polytopes. Naïvely, we can obtain a convex-combination representation by the following algorithm.

Algorithm 13.

$t \leftarrow 1$;
while $E(\boldsymbol{x}_{t-1}) \neq \emptyset$ **do**
  begin (**comment:** we call the following block as t-th iteration)
    find a perfect matching M_t of the bipartite graph $(U, V; E(\boldsymbol{x}_{t-1}))$;
    $\lambda(M_t) \leftarrow \min\{x(e) \mid e \in M_t\}$;
    $\boldsymbol{x}_t \leftarrow \boldsymbol{x}_{t-1} - \lambda(M_t)\boldsymbol{\chi}(M_t)$;
    $t \leftarrow t + 1$
  end;
$T \leftarrow t - 1$;
return the representation $\boldsymbol{x}_0 = \sum_{t=1}^{T} \lambda(M_t)\boldsymbol{\chi}(M_t)$.

For each iteration, there is a perfect matching in $(U, V; E(\boldsymbol{x}_{t-1}))$, according to Proposition 12.

Initially the bipartite graph $(U, V; E(\boldsymbol{x}_0))$ has $|E(\boldsymbol{x}_0)| = \mathrm{O}(n^2)$ edges. At t-th iteration, nonempty edge set $\{e \in M_t \mid x(e) = \lambda(M_t)\}$ is removed. Thus the total number of iterations is bounded by $\mathrm{O}(n^2)$. In each iteration, Hopcroft–Karp's algorithm finds a perfect matching in $\mathrm{O}(n^{2.5})$ time [14] and other operations require $\mathrm{O}(n^2)$ time. Therefore, the time complexity of Algorithm 13 is bounded by $\mathrm{O}(n^{4.5})$.

The following algorithm reduces the computational time to $\mathrm{O}(n^4)$.

Algorithm 14.

$t \leftarrow 1$, $M'_0 \leftarrow \emptyset$;
while $E(\boldsymbol{x}_{t-1}) \neq \emptyset$ **do**
  begin (**comment:** we call the following block as t-th iteration)
    using the bipartite graph $(U, V; E(\boldsymbol{x}_{t-1}))$ and the matching M'_{t-1}, find a perfect matching M_t by the augmenting-path method;
    $\lambda(M_t) \leftarrow \min\{x(e) \mid e \in M_t\}$;
    $\boldsymbol{x}_t \leftarrow \boldsymbol{x}_{t-1} - \lambda(M_t)\boldsymbol{\chi}(M_t)$;
    $M'_t \leftarrow M_t \cap E(\boldsymbol{x}_t)$;
    $t \leftarrow t + 1$
  end;
$T \leftarrow t - 1$;

return the representation $\boldsymbol{x}_0 = \sum_{t=1}^{T} \lambda(M_t)\boldsymbol{\chi}(M_t)$.

According to Proposition 12, $(U, V; E(\boldsymbol{x}_{t-1}))$ has a perfect matching.

The augmenting-path method increases the cardinality of matching by exactly one in $\mathrm{O}(n^2)$ time (see [21: pp. 263–264], for example). We should estimate how many times we execute the augmenting-path method totally. When $t = 1$, because we need to produce a perfect matching from the empty set, we apply the augmenting-path method n times. Between $t = 2$ and T, we need to supply edges at t-th iteration as many as the edges lost on making M'_{t-1} from M_{t-1}. The number of lost edges is equal to $|\{e \in M_{t-1} \mid x(e) = \lambda(M_{t-1})\}| = |E(\boldsymbol{x}_{t-2})| - |E(\boldsymbol{x}_{t-1})|$. By summing up the required number of executions of the augmenting-path method from $t = 2$ to T, we obtain that

$$\sum_{t=2}^{T} (|E(\boldsymbol{x}_{t-2})| - |E(\boldsymbol{x}_{t-1})|) = |E(\boldsymbol{x}_0)| - |E(\boldsymbol{x}_{T-1})| \leq |E(\boldsymbol{x}_0)| = \mathrm{O}(n^2).$$

From the above, we call the augmenting-path method $\mathrm{O}(n^2)$ times through the algorithm. Except for the augmenting-path method step, procedures require at most $\mathrm{O}(n^2)$ time for each iteration and the total number of iterations is bounded by $\mathrm{O}(n^2)$. Consequently, Algorithm 14 requires $\mathrm{O}(n^4)$ time.

References

[1] Alizadeh, F., Goldfarb, D.: Second-order cone programming. *Mathematical Programming*, **95** (2003), 3–51.

[2] Bandelt, H.-J., Crama, Y., Spieksma, F. C. R.: Approximation algorithms for multi-dimensional assignment problems with decomposable costs. *Discrete Applied Mathematics*, **49** (1994), 25–50.

[3] Birkhoff, G.: Tres observaciones sobre el algebra lineal. *Universidad Nacionale Tucumán Revista*, **5** (1946), 147–151.

[4] Boros, E., Hammer, P. L.: On clustering problems with connected optima in Euclidean spaces. *Discrete Mathematics*, **75** (1989), 81–88.

[5] Burkard, R. E., Rudolf, R., Woeginger, G. J.: Three-dimensional axial assignment problems with decomposable cost coefficients. *Discrete Applied Mathematics*, **65** (1996), 123–139.

[6] Burkard, R. E., Çela, E.: Linear assignment problems and extensions. In: Du, D.-Z., Pardalos, P. M. (eds.): *Handbook of Combinatorial Optimization, Supplement Volume A*, 1999, Kluwer Academic Publishers, 75–149.

[7] Capoyleas, V., Rote, G., Woeginger, G.: Geometric clusterings. *Journal of Algorithms*, **12** (1991), 341–356.

[8] Crama, Y., Kolen, A. W. J., Oerlemans, A. G., Spieksma, F. C. R.: Throughput rate optimization in the automated assembly of printed circuit boards. *Annals of Operations Research*, **26** (1991), 455–480.

[9] Crama, Y., Spieksma, F. C. R.: Approximation algorithms for three-dimensional assignment problems with triangle inequalities. *European Journal of Operational Research*, **60** (1992), 273–279.

[10] Dyer, M. E., Frieze, A. M.: Planar 3DM is NP-complete. *Journal of Algorithms*, **7** (1986), 174–184.

[11] Frieze, A. M., Yadegar, J.: An algorithm for solving 3-dimensional assignment problems with application to scheduling a teaching practice. *Journal of the Operational Research Society*, **32** (1981), 989–995.

[12] Garey, M. R., Johnson, D. S.: *Computers and Intractability: A Guide to the Theory of NP Completeness*. 1979, W. H. Freeman.

[13] Haley, K. B.: The multi-index problem. *Operations Research*, **11** (1963), 368–379.

[14] Hopcroft, J. E., Karp, R. M.: An $n^{5/2}$ algorithm for maximum matching in bipartite graphs. *SIAM Journal on Computing*, **2** (1973), 225–231.

[15] von Neumann, J.: A certain zero-sum two-person game equivalent to the optimal assignment problem. *Contributions to the Theory of Games*, **2** (1953), 5–12.

[16] Pierskalla, W. P.: The tri-substitution method for the three-dimensional assignment problem. *Journal of the Canadian Operational Research Society*, **5** (1967), 71–81.

[17] Pierskalla, W. P.: The multidimensional assignment problem. *Operations Research*, **16** (1968), 422–431.

[18] Poore, A. B.: Multidimensional assignment formulation of data association problems arising from multitarget and multisensor tracking. *Computational Optimization and Applications*, **3** (1994), 27–54.

[19] Poore, A. B., Rijavec, N.: Partitioning multiple data sets: multidimensional assignments and Lagrangian relaxation. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, **16** (1994), 317–342.

[20] Rässler, S.: *Statistical Matching*. Lecture Notes in Statistics **168**, 2002, Springer.

[21] Scrijver, A.: *Combinatorial Optimization*. Algorithms and Combinatorics **24**, 2003, Springer.

[22] Soong, R., de Montigny, M.: No free lunch in data fusion/integration. *Session Papers of Worldwide Audience Measurement*, June 2004, Geneva, 33–54.

[23] Spieksma, F. C. R., Woeginger, G. J.: Geometric three-dimensional assignment problems. *European Journal of Operational Research*, **91** (1996), 611–618.

[24] Spieksma, F. C. R.: Multi index assignment problems: complexity, approximation, applications. In: Pardalos, P. M., Pitsoulis, L. S. (eds.): *Nonlinear Assignment Problems, Algorithms and Applications*, 2000, Kluwer Academic Publishers, 1–12.

[25] Tamassia, R., Tollis, I. G.: Planar grid embedding in linear time. *IEEE Transactions on circuits and systems*, **39** (1989), 1230–1234.