Department of Social Systems and Management

Discussion Paper Series

No.1271

Lagrangian Relaxation and Pegging Test for Linear Ordering Problems

by

Noriyoshi SUKEGAWA, Yoshitsugu YAMAMOTO and Liyuan ZHANG

May 2011

UNIVERSITY OF TSUKUBA
Tsukuba, Ibaraki 305-8573
JAPAN

LAGRANGIAN RELAXATION AND PEGGING TEST FOR LINEAR ORDERING PROBLEMS

NORIYOSHI SUKEGAWA, YOSHITSUGU YAMAMOTO, AND LIYUAN ZHANG

Abstract. We develop an algorithm for the linear ordering problem (LOP), which has a large number of applications such as triangulation of input-output matrices, minimizing total weighted completion time in one-machine scheduling, and aggregation of individual preferences. The algorithm is based on the Lagrangian relaxation of the binary integer linear programming formulation. Since the number of the constraints that should be relaxed is proportional to the cubed number of items and too large to handle, we propose a modified subgradient method that ignores a part of the constraints and gradually adds constraints whose Lagrangian multiplier vector is likely to be positive at the optimal solution. We also propose an improvement of the ordinary pegging test by using the problem structure. The information obtained from the improved pegging test is used in an attempt to get a good incumbent in an early stage of computation.

1. Introduction

The problem we consider in this paper is to find a linear ordering of n items when their pairwise comparison data is given. The data is given by an $n \times n$ matrix $C := [c_{ij}]$ such that its (i, j)th element carries the amount of profit made when item i is ranked prior to item j. Choosing an appropriate matrix C lets the problem embrace ranking aggregation problem, which is called Kemeny's problem, minimum violations ranking problem, and Slater's problem. See the survey paper by Charon and Hudry [8] and Reinelt [23]. The problem is formulated as a linear integer programming problem. The polytope being the convex hull of binary vectors each corresponding to a linear ordering was named *linear ordering polytope* and investigated by Grötschel et al. [16]. They introduced some facet-defining valid inequalities of the polytope, and proposed a linear-programming-relaxation-based algorithm for the problem in [15]. For subsequent research on the linear ordering polytope, see [4, 10, 18, 21]. Their approach was further extended by Mitchell and Borchers [19, 20], who proposed a cutting plane algorithm based on a primal-dual interior point method, and solved problems with as many as 250 items. Since the problem is an NP-hard problem, see e.g., Section 2 of [8], there have been proposed several heuristic methods, e.g., Lagrangian heuristic method in [3], scatter search method in [5], linear ordering construction heuristics in [9], Goddard's method in [13], variable neighborhood local search method in [14]. Charon and Hudry [7] made an experiment of a branch-and-bound method with Lagrangian relaxation and some heuristics.

The binary integer programming formulation of the linear ordering problem has an $O(n^3)$ of inequality constraints. This feature makes the problem hard to solve. In this paper, we propose a Lagrangian relaxation algorithm that considers a small fraction of the inequality constraits and a pegging test that takes advantage of the problem structure. The algorithm is a combination of well-known and widely-used techniques of mathematical

Date: May 10, 2011.

Key words and phrases. linear ordering problem, Lagrangian relaxation, Lagrangian dual, subgradient method, pegging test, duality gap, linear ordering polytope.

The authors thank A.N. Langville and K. Pedings for providing the data. The second author is partly supported by Grant-in-Aid for Scientific Research (C) 22510136.

optimization, however, it would be well worth reporting how they function together and how they make the algorithm efficient in an integrated manner.

Throughout this paper we will use the following symbols:

$$\begin{array}{ll} N := \{1, 2, \dots, n\} & N^2 := \{\, (i,j) \mid i,j \in N \,\} \\ N^2_{\neq} := \{\, (i,j) \mid i,j \in N, i \neq j \,\} & N^3_{\neq} := \{\, (i,j,k) \mid i,j,k \in N, i \neq j, j \neq k, k \neq i \,\} \\ N^2_{<} := \{\, (i,j) \mid i,j \in N, i < j \,\} & N^3_{<} := \{\, (i,j,k) \mid i,j,k \in N, i < j < k \,\}. \end{array}$$

2. Linear Ordering Problem

2.1. **Ranking aggregation.** Suppose we have several different rankings of n items, and want to aggregate them to a single ranking or a linear ordering. If each ranking comes from the ratings of items, summing up the ratings that item i receives to its overall rating and sorting them for a final linear ordering is a possible and widely used method. Our starting point in this section is not the ratings of items but their rankings. One of the well-known method for aggregation of rankings is the Borda method which was first proposed in the 18th century. As Kemeny proposed in [17], a natural solution would be a linear ordering that is "close" to all given rankings. Let $\sigma_1, \dots, \sigma_\kappa, \dots, \sigma_K$ be given rankings of items N. Let for $\alpha \in [0,1]$

$$c^{(1)}_{ij} := \alpha \big| \{\, \kappa \mid \sigma_\kappa(i) < \sigma_\kappa(j) \,\} \big| - (1-\alpha) \big| \{\, \kappa \mid \sigma_\kappa(i) > \sigma_\kappa(j) \,\} \big| \tag{2.1}$$

$$c^{(2)}_{ij} := \sum_{\kappa=1}^{K} \alpha \, [\sigma_\kappa(j) - \sigma_\kappa(i)]^+ - (1-\alpha) \, [\sigma_\kappa(i) - \sigma_\kappa(j)]^+ \,, \tag{2.2}$$

where $|\cdot|$ denotes the cardinality of the corresponding set, and $[t]^+ = \max\{t, 0\}$. The coefficient $c^{(1)}_{ij}$ is a weighted difference of the number of rankings that put i above j and those that put i below j, and $c^{(2)}_{ij}$ is the weighted sum of differences between the rankings of i and j. The parameter α should be determined according to which of "aye" and "nay" is more important. When $\alpha = 1/2$, $c^{(2)}_{ij} = (1/2) \sum_{\kappa=1}^{K} (\sigma_\kappa(j) - \sigma_\kappa(i))$.

Let π denote an aggregated linear ordering. The values $c^{(\nu)}_{ij}$ for $\nu = 1, 2$ shows how the linear ordering π and given rankings agree about the order of i and j when $\pi(i) < \pi(j)$. Hence the overall degree of agreement would be given by

$$\sum_{(i,j):\pi(i)<\pi(j)} c^{(\nu)}_{ij}.$$

A linear ordering that maximizes this function should be accepted as the "closest" aggregated linear ordering, hence our problem is formulated as

$$(LOP) \quad \left| \begin{array}{ll} \text{maximize} & \displaystyle\sum_{(i,j):\pi(i)<\pi(j)} c^{(\nu)}_{ij} \\ \text{subject to} & \pi \text{ is a linear ordering,} \end{array} \right.$$

which we will refer to as the *Linear Ordering Problem.*

2.2. **Minimum violations ranking.** Ali et al. [1] and Pedings et al. [22] proposed *minimum violations ranking.* Suppose we are given a matrix $D := [d_{ij}]_{(i,j) \in N^2}$ such that d_{ij} is the points by which team i beats team j in their matchup, where we take the convention that $d_{ii} = 0$. They call this matrix a *point differential matrix* and introduce the following definition.

Definition 2.1. A matrix D is in *hillside form* if

$$\begin{aligned} d_{ki} &\leq d_{kj} \quad \text{(ascending order across rows)} \\ d_{ik} &\geq d_{jk} \quad \text{(descending order down columns)} \end{aligned}$$

for all $i, j, k \in N^3_{\neq}$ such that $i < j$.

They proposed to find such a hidden hillside form by a simultaneous reordering of rows and columns of the given point differential matrix D, and showed that the problem is formulated[1] as (LOP) in the previous section with the following objective function coefficient for $\alpha = 1/2$

$$c^{(3)}_{ij} := \alpha|\{\, k \in N \setminus \{i, j\} \mid d_{ki} \leq d_{kj} \,\}| + (1 - \alpha)|\{\, k \in N \setminus \{i, j\} \mid d_{ik} \geq d_{jk} \,\}|. \tag{2.3}$$

Another choice of the objective function coefficient would be

$$c^{(4)}_{ij} := \sum_{k \in N \setminus \{i,j\}} \alpha(d_{kj} - d_{ki}) + (1 - \alpha)(d_{ik} - d_{jk}). \tag{2.4}$$

3. Formulation of Linear Ordering Problem

3.1. **Quadratic Assignment Formulation.** Let y_{ki} be the binary variable such that

$$y_{ki} = \begin{cases} 1 & \text{if } k\text{th ranking is given to item } i \\ 0 & \text{otherwise.} \end{cases}$$

Then these variables satisfy

$$\sum_{k \in N} y_{ki} = 1 \quad \text{for all } i \in N, \tag{3.1}$$

$$\sum_{i \in N} y_{ki} = 1 \quad \text{for all } k \in N. \tag{3.2}$$

The cost concerning the ordered pair (i, j) is

$$\begin{aligned} & c_{ij}\big(y_{1i}y_{2j} + y_{1i}y_{3j} + \cdots + y_{1i}y_{nj} + y_{2i}y_{3j} + y_{2i}y_{4j} + \cdots + y_{2i}y_{nj} + \cdots + y_{(n-1)i}y_{nj}\big) \\ &= c_{ij}\big(y_{1i}(y_{2j} + y_{3j} + \cdots + y_{nj}) + y_{2i}(y_{3j} + y_{4j} + \cdots + y_{nj}) + \cdots + y_{(n-1)i}y_{nj}\big) \\ &= c_{ij} \sum_{k=1}^{n-1} y_{ki}\Big(\sum_{l=k+1}^{n} y_{lj}\Big) \end{aligned}$$

and the total agreement is given by

$$\sum_{(i,j) \in N^2_{\neq}} c_{ij} \sum_{k=1}^{n-1} y_{ki}\Big(\sum_{l=k+1}^{n} y_{lj}\Big).$$

The quadratic assignment formulation is to maximize the total agreement under the assignment constraints (3.1) and (3.2) together with the binary variable constraints. This is a well-known *NP*-hard problem and already a challenging problem when $n = 25$. See Çela [6].

[1] To be very precise, Pedings et al. formulated the problem as a minimization of violations of the hillside structure.

3.2. **Integer Linear Programming Formulation.** The coefficient c_{ij} provides the degree of agreement of a linear ordering π such that $\pi(i) < \pi(j)$. For a given linear ordering π let binary variables x_{ij} for $(i,j) \in N^2_{\neq}$ be defined as

$$x_{ij} = \begin{cases} 1 & \text{if } \pi(i) < \pi(j) \\ 0 & \text{otherwise,} \end{cases}$$

then the linear ordering problem is formulated as

$$(LOP) \quad \left| \begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_{\neq}} c_{ij}x_{ij} & \\ \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_{\neq} \quad \text{(binary)} \\ & x_{ij} + x_{ji} = 0 & \text{for all } (i,j) \in N^2_{\neq} \quad \text{(antisymmetry)} \\ & x_{ij} + x_{jk} + x_{ki} \leq 2 & \text{for all } (i,j,k) \in N^3_{\neq} \quad \text{(transitivity).} \end{array} \right.$$

The point is that the problem has $n(n-1)$ binary variables, $n(n-1)/2$ equality constraints and $n(n-1)(n-2)/3$ inequality constraints, all of which grow very rapidly as n grows.

3.3. **Variable reduction.** Substituting $1 - x_{ij}$ for x_{ji} for all $i, j \in N$ with $i < j$ halves the decision variables and yields the following equivalent problem (P):

$$(P) \quad \left| \begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_{<}} \bar{c}_{ij}x_{ij} + \sum_{(i,j)\in N^2_{<}} c_{ji} & \\ \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_{<} \\ & x_{ij} + x_{jk} - x_{ik} \leq 1 & \text{for all } (i,j,k) \in N^3_{<} \quad \text{(type 1)} \\ & -x_{ij} - x_{jk} + x_{ik} \leq 0 & \text{for all } (i,j,k) \in N^3_{<} \quad \text{(type 2),} \end{array} \right.$$

where

$$\bar{c}_{ij} := c_{ij} - c_{ji}.$$

We will call the inequality constraint of the first half the *tranisitivity constraint of type 1*, and one of the latter half *type 2*, and we will denote the optimal objective function value of (P) by $\omega(P)$.

4. Relaxation

4.1. **Relaxation of inequality constraints.** A possible relaxation is to temporarily discard some of the inequality constraints. Namely let U and V be subsets of $N^3_{<}$ and solve

$$(P(U,V)) \quad \left| \begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_{<}} \bar{c}_{ij}x_{ij} & \\ \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_{<} \\ & x_{ij} + x_{jk} - x_{ik} \leq 1 & \text{for all } (i,j,k) \in U \\ & -x_{ij} - x_{jk} + x_{ik} \leq 0 & \text{for all } (i,j,k) \in V. \end{array} \right.$$

Clearly if the optimal solution of $(P(U,V))$ satisfies all the transitivity constraints, it is an optimal solution of problem (P).

4.2. **Lagrangian Relaxation.** Problem $(P(U,V))$ is still a difficult problem to solve unless no favorable structure can be assumed on U and V. One of the common tricks to deal with the problem would be the Lagrangian relaxation. Namely, introducing a nonnegative multiplier u_{ijk} for each constraint of type 1 and also a nonnegative multiplier v_{ijk} for each constraint of type 2, we consider the following integer linear programming with only a simple binary variable constraint:

$$(LR(\boldsymbol{u},\boldsymbol{v}))\quad \left|\begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_<} \bar{c}_{ij}x_{ij} & +\displaystyle\sum_{(i,j,k)\in U} u_{ijk}(1-x_{ij}-x_{jk}+x_{ik}) \\ & & +\displaystyle\sum_{(i,j,k)\in V} v_{ijk}(0+x_{ij}+x_{jk}-x_{ik}) \\ \text{subject to} & x_{ij}\in\{0,1\} & \text{for all } (i,j)\in N^2_<. \end{array}\right.$$

Omitting U and V, we denote this problem simply by $(LR(\boldsymbol{u},\boldsymbol{v}))$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ denote multiplier vectors $(u_{ijk})_{(i,j,k)\in U}$ and $(v_{ijk})_{(i,j,k)\in V}$, respectively. Let $r(\boldsymbol{u},\boldsymbol{v})_{ij}$ denote the coefficient of variable x_{ij} in the objective function. It is written as

$$\begin{aligned} r(\boldsymbol{u},\boldsymbol{v})_{ij} = \bar{c}_{ij} &- \sum_{k:(i,j,k)\in U} u_{ijk} - \sum_{k:(k,i,j)\in U} u_{kij} + \sum_{k:(i,k,j)\in U} u_{ikj} \\ &+ \sum_{k:(i,j,k)\in V} v_{ijk} + \sum_{k:(k,i,j)\in V} v_{kij} - \sum_{k:(i,k,j)\in V} v_{ikj}. \end{aligned} \tag{4.1}$$

Due to the simple constraint, an optimal solution $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v}) = (x(\boldsymbol{u},\boldsymbol{v})_{ij})_{(i,j)\in N^2_<}$ of problem $(LR(\boldsymbol{u},\boldsymbol{v}))$ can be obtained by

$$x(\boldsymbol{u},\boldsymbol{v})_{ij} = \begin{cases} 1 & \text{if } r(\boldsymbol{u},\boldsymbol{v})_{ij} > 0 \\ 0 & \text{if } r(\boldsymbol{u},\boldsymbol{v})_{ij} \le 0. \end{cases} \tag{4.2}$$

Furthermore, the optimal objective function value, which we will denote by $\omega(LR(\boldsymbol{u},\boldsymbol{v}))$, provides an upper bound of the optimal objective function value $\omega(P)$ of problem (P).

5. Optimality and Duality Gap

The following theorem is well known, see e.g., Geoffrion [12].

Theorem 5.1. *Let $(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}}) := ((\bar{u}_{ijk})_{(i,j,k)\in N^3_<}, (\bar{v}_{ijk})_{(i,j,k)\in N^3_<})$ be a Lagrangian multiplier vector corresponding to all the transitivity constraints, and let $\boldsymbol{x}$ be an optimal solution of the Lagrangian relaxation problem of (P) with $(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}})$. If $\boldsymbol{x}$ is feasible to problem (P) and satisfies the complementarity condition*

$$\begin{aligned} \bar{u}_{ijk}(1-x_{ij}-x_{jk}+x_{ik}) = 0 \quad &\textit{for all } (i,j,k)\in N^3_< \\ \bar{v}_{ijk}(0+x_{ij}+x_{jk}-x_{ik}) = 0 \quad &\textit{for all } (i,j,k)\in N^3_<, \end{aligned}$$

then it is an optimal solution of (P).

Definition 5.2. We say that $\boldsymbol{x}$ satisfies the *restricted complementarity condition with* $(\boldsymbol{u},\boldsymbol{v})$ when

$$\begin{aligned} u_{ijk}(1-x_{ij}-x_{jk}+x_{ik}) = 0 \quad &\text{for all } (i,j,k)\in U \\ v_{ijk}(0+x_{ij}+x_{jk}-x_{ik}) = 0 \quad &\text{for all } (i,j,k)\in V, \end{aligned}$$

We readily see the following corollary.

Corollary 5.3. *If an optimal solution $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ of $(LR(\boldsymbol{u},\boldsymbol{v}))$ is feasible to problem (P) and satisfies the restricted complementarity condition with $(\boldsymbol{u},\boldsymbol{v})$, then it is an optimal solution of (P).*

Proof. We readily see that the Lagrangian relaxation problem $(LR(\boldsymbol{u},\boldsymbol{v}))$ is an ordinary Lagrangian relaxation problem of problem (P) with multipliers $(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}})$ such that

$$\bar{u}_{ijk} = \begin{cases} u_{ijk} & \text{for } (i,j,k) \in U \\ 0 & \text{for } (i,j,k) \in N^3_< \setminus U \end{cases} \qquad \bar{v}_{ijk} = \begin{cases} v_{ijk} & \text{for } (i,j,k) \in V \\ 0 & \text{for } (i,j,k) \in N^3_< \setminus V. \end{cases}$$

When $x(\boldsymbol{u},\boldsymbol{v})$ meets the restricted complementarity condition with $(\boldsymbol{u},\boldsymbol{v})$ in Definition 5.2, it also satisfies the complementarity condition for all constraints with $(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}})$. This together with the feasibility of $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ yields the desired result. □

A feasible solution of problem (P) that has the largest objective function value among the feasible solutions found thus far is called an *incumbent solution*, and its objective function value is called an *incumbent value*. The difference of $\omega(LR(\boldsymbol{u},\boldsymbol{v}))$ and the incumbent value is called the *duality gap*.

6. Pegging Test

6.1. **Ordinary pegging test.** By the information obtained from the optimal solution $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ of the Lagrangian relaxation problem $(LR(\boldsymbol{u},\boldsymbol{v}))$ we can see which variable takes one and which takes zero at the optimal solution of problem (P). Let us choose $(s,t) \in N^2_<$ and suppose that problem (P) has an optimal solution with $x_{st} = \xi$ for some $\xi \in \{0,1\}$. Then problem (P) with an additional constraint $x_{st} = \xi$ is equivalent to problem (P) in the sense that optimal values of the two problems coincide. Suppose further we have an incumbent value ω_{low}. Then clearly

$$\omega(P|x_{st} = \xi) = \omega(P) \geq \omega_{\text{low}}.$$

Since $(P(U,V))$ is a relaxation of problem (P), and it is further relaxed to $(LR(\boldsymbol{u},\boldsymbol{v}))$, we obtain

$$\omega(LR(\boldsymbol{u},\boldsymbol{v})|x_{st} = \xi) \geq \omega(P(U,V)|x_{st} = \xi) \geq \omega(P|x_{st} = \xi),$$

hence

$$\omega(LR(\boldsymbol{u},\boldsymbol{v})|x_{st} = \xi) \geq \omega_{\text{low}}.$$

Lemma 6.1. *Let ξ be either zero or one. If $\omega(LR(\boldsymbol{u},\boldsymbol{v})|x_{st} = \xi) < \omega_{\text{low}}$, then $x_{st} = 1-\xi$ for any optimal solution of problem (P).*

Proof. Straightforward from the above discussion. □

Suppose that we have an optimal solution $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ of $(LR(\boldsymbol{u},\boldsymbol{v}))$ and that $x(\boldsymbol{u},\boldsymbol{v})_{st} = 0$. By a simple calculation we see that

$$\omega(LR(\boldsymbol{u},\boldsymbol{v})|x_{st} = 1) = \omega(LR(\boldsymbol{u},\boldsymbol{v})) + r(\boldsymbol{u},\boldsymbol{v})_{st}. \tag{6.1}$$

Note that $x(\boldsymbol{u},\boldsymbol{v})_{st} = 0$ implies $r(\boldsymbol{u},\boldsymbol{v})_{st} \leq 0$. In the same way we see that

$$\omega(LR(\boldsymbol{u},\boldsymbol{v})|x_{st} = 0) = \omega(LR(\boldsymbol{u},\boldsymbol{v})) - r(\boldsymbol{u},\boldsymbol{v})_{st} \tag{6.2}$$

when $x(\boldsymbol{u},\boldsymbol{v})_{st} = 1$. Note also that $r(\boldsymbol{u},\boldsymbol{v})_{st} > 0$ in this case.

Theorem 6.2. *Let $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ be an optimal solution of the Lagrangian relaxation problem $(LR(\boldsymbol{u},\boldsymbol{v}))$. if*

$$\omega(LR(\boldsymbol{u},\boldsymbol{v})) - \omega_{\text{low}} < |r(\boldsymbol{u},\boldsymbol{v})_{st}|$$

holds, then $x^*_{st} = x(\boldsymbol{u},\boldsymbol{v})_{st}$ for any optimal solution $\boldsymbol{x}^*$ of (P).

Proof. Substituting equation (6.1) or (6.2) for the condition in Lemma 6.1 will yield the assertion. □

We say that the variable x_{st} is *pegged* at $x(\boldsymbol{u},\boldsymbol{v})_{st}$ when the case holds in the theorem.

6.2. **Improved Pegging Test.** As the computation goes, we will have several variables pegged. Let P_0 and P_1 denote the index sets of the variables that have been pegged at zero and one, respectively. Given a Lagrangian multiplier vector $(\boldsymbol{u}, \boldsymbol{v})$, the problem

$$(LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1)) \quad \left|\begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_<} \bar{c}_{ij}x_{ij} & + \displaystyle\sum_{(i,j,k)\in U} u_{ijk}(1 - x_{ij} - x_{jk} + x_{ik}) \\ & & + \displaystyle\sum_{(i,j,k)\in V} v_{ijk}(0 + x_{ij} + x_{jk} - x_{ik}) \\ \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_< \\ & x_{ij} = \begin{cases} 0 \\ 1 \end{cases} & \begin{array}{l} \text{for all } (i,j) \in P_0 \\ \text{for all } (i,j) \in P_1. \end{array} \end{array}\right.$$

is a relaxation problem of (P).

Let $A(P_0, P_1)$ be the set of arcs (i, j) such that either x_{ij} has been pegged at one or x_{ji} has been pegged at zero, i.e.,

$$A(P_0, P_1) := \{\, (i,j) \in N^2_{\neq} \mid (j,i) \in P_0 \text{ or } (i,j) \in P_1 \,\}. \tag{6.3}$$

Definition 6.3. Given P_0 and P_1 and $i, j \in N$, we say that i is an *ancestor* of j and also that j is a *descendant* of i when there is a directed path from i to j on the arc set $A(P_0, P_1)$.

Definition 6.4. Given $(s,t) \in N^2_< \setminus (P_0 \cup P_1)$ let

$$S_1 := \{s\} \cup \{\, i \in N \mid i \text{ is an ancestor of } s \,\}, \quad T_1 := \{t\} \cup \{\, j \in N \mid j \text{ is a descendant of } t \,\}$$
$$S_0 := \{s\} \cup \{\, i \in N \mid i \text{ is a descendant of } s \,\}, \quad T_0 := \{t\} \cup \{\, j \in N \mid j \text{ is an ancestor of } t \,\}.$$

Take a variable x_{st} that has not yet been pegged, i.e., $(s,t) \in N^2_< \setminus (P_0 \cup P_1)$, and fix x_{st} temporarily to one. Then every ancestor of s should be an ancestor of every descendant of t by the transitivity. Namely, the variables must satisfy

$$x_{ij} = \begin{cases} 1 & \text{for all } (i,j) \in (S_1 \times T_1) \cap N^2_< \\ 0 & \text{for all } (i,j) \in (T_1 \times S_1) \cap N^2_< \end{cases} \tag{6.4}$$

to meet the transitivity constraint. When x_{st} is fixed temporarily to zero, we have similarly

$$x_{ij} = \begin{cases} 1 & \text{for all } (i,j) \in (T_0 \times S_0) \cap N^2_< \\ 0 & \text{for all } (i,j) \in (S_0 \times T_0) \cap N^2_<. \end{cases} \tag{6.5}$$

Now given nonnegative multiplier vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ we define

$$\begin{array}{r} (LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1) \\ |x_{st} = 1) \end{array} \quad \left|\begin{array}{lll} \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_<} \bar{c}_{ij}x_{ij} & + \displaystyle\sum_{(i,j,k)\in U} u_{ijk}(1 - x_{ij} - x_{jk} + x_{ik}) \\ & & + \displaystyle\sum_{(i,j,k)\in V} v_{ijk}(0 + x_{ij} + x_{jk} - x_{ik}) \\ \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_< \\ & x_{ij} = \begin{cases} 0 \\ 1 \end{cases} & \begin{array}{l} \text{for all } (i,j) \in (T_1 \times S_1) \cap N^2_< \cup P_0 \\ \text{for all } (i,j) \in (S_1 \times T_1) \cap N^2_< \cup P_1. \end{array} \end{array}\right.$$

sophisticate This is a relaxation problem of (P) with a temporary constraint $x_{st} = 1$ added.

Lemma 6.5. *If* $((S_1 \times T_1) \cap N^2_< \cap P_0) \cup ((T_1 \times S_1) \cap N^2_< \cap P_1) \neq \emptyset$*, then* $x^*_{st} = 0$ *for any optimal solution* $\boldsymbol{x}^*$ *of problem* (P).

Proof. If there is an element, say (i,j), in the set $((S_1 \times T_1) \cap N^2_< \cap P_0) \cup ((T_1 \times S_1) \cap N^2_< \cap P_1)$, the variable x_{ij} must be zero and one at the same time, which implies that there is no optimal solution of (P) with $x_{st} = 1$. $\square$

When x_{st} is temporarily fixed to zero, we have the following problem and lemma.

$$
\begin{array}{r|lll}
 & \text{maximize} & \displaystyle\sum_{(i,j)\in N^2_<} \bar{c}_{ij} x_{ij} + \sum_{(i,j,k)\in U} u_{ijk}(1 - x_{ij} - x_{jk} + x_{ik}) & \\
(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1) & & \displaystyle\quad + \sum_{(i,j,k)\in V} v_{ijk}(0 + x_{ij} + x_{jk} - x_{ik}) & \\
|x_{st} = 0) & \text{subject to} & x_{ij} \in \{0,1\} & \text{for all } (i,j) \in N^2_< \\
 & & x_{ij} = \begin{cases} 0 \\ 1 \end{cases} & \begin{array}{l} \text{for all } (i,j) \in (S_0 \times T_0) \cap N^2_< \cup P_0 \\ \text{for all } (i,j) \in (T_0 \times S_0) \cap N^2_< \cup P_1. \end{array}
\end{array}
$$

Lemma 6.6. *If $((T_0 \times S_0) \cap N^2_< \cap P_0) \cup ((S_0 \times T_0) \cap N^2_< \cap P_1) \neq \emptyset$, then $x^*_{st} = 1$ for any optimal solution $\boldsymbol{x}^*$ of problem (P).*

Since the problem $(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1)|x_{st} = \xi)$ is a relaxation problem of (P) with a constraint $x_{st} = \xi$ added, we readily see the following lemma.

Lemma 6.7. *Let ξ be either zero or one, and let x_{st} be a variable that has not been pegged, i.e., $(s,t) \in N^2_< \setminus (P_0 \cup P_1)$. If $\omega(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1)|x_{st} = \xi) < \omega_{\text{low}}$, then $x_{st} = 1 - \xi$ for any optimal solution of problem (P).*

We have seen in (6.1) and (6.2) that

$$\omega(LR(\boldsymbol{u}, \boldsymbol{v})) - \omega(LR(\boldsymbol{u}, \boldsymbol{v})|x_{st} = 1 - x(\boldsymbol{u}, \boldsymbol{v})_{st}) = |r(\boldsymbol{u}, \boldsymbol{v})_{st}|$$

holds. Namely, the objective function value deteriorates by $|r(\boldsymbol{u}, \boldsymbol{v})_{st}|$ when the additional constraint $x_{st} = 1 - x(\boldsymbol{u}, \boldsymbol{v})_{st}$ is added to $(LR(\boldsymbol{u}, \boldsymbol{v}))$. In the similar manner we see that

$$\omega(LR(\boldsymbol{u}, \boldsymbol{v})) - \omega(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1)|x_{st} = 1 - x(\boldsymbol{u}, \boldsymbol{v})_{st}) = \sum_{(i,j)\in D} |r(\boldsymbol{u}, \boldsymbol{v})_{ij}|,$$

where

$$
\begin{aligned}
D = & (((S_1 \times T_1) \cap N^2_< \cup P_1) \cap \{(i,j) \mid x(\boldsymbol{u}, \boldsymbol{v})_{ij} = 0\}) \\
& \cup (((T_1 \times S_1) \cap N^2_< \cup P_0) \cap \{(i,j) \mid x(\boldsymbol{u}, \boldsymbol{v})_{ij} = 1\}) \\
& \text{when } x(\boldsymbol{u}, \boldsymbol{v})_{st} = 0, \text{and} \\
D = & (((T_0 \times S_0) \cap N^2_< \cup P_1) \cap \{(i,j) \mid x(\boldsymbol{u}, \boldsymbol{v})_{ij} = 0\}) \\
& \cup (((S_0 \times T_0) \cap N^2_< \cup P_0) \cap \{(i,j) \mid x(\boldsymbol{u}, \boldsymbol{v})_{ij} = 1\}) \\
& \text{when } x(\boldsymbol{u}, \boldsymbol{v})_{st} = 1.
\end{aligned}
\tag{6.6}
$$

The first subset of D corresponds to the variables that should be one but takes zero at $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$, and the second subset to those that should be zero but takes one at $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$.

6.3. **Transitive Closure.** As was seen in the previous section, it would be useful and save computation time to peg as many variables as possible. This can be done by computing the transitive closure of the directed graph consisting of node set N and arc set $A(P_0, P_1)$ of (6.3). The *transitive closure* of $(N, A(P_0, P_1))$ is a directed graph $(N, \bar{A})$ such that $(i,j) \in \bar{A}$ if and only if there is a directed path from i to j in $A(P_0, P_1)$. Once we have made the transitive closure, the sets used in the improved pegging test are readily obtained by

$$
\begin{aligned}
S_1 &:= \{s\} \cup \{\, i \in N \mid (i,s) \in \bar{A} \,\}, \quad T_1 := \{t\} \cup \{\, j \in N \mid (t,j) \in \bar{A} \,\} \\
S_0 &:= \{s\} \cup \{\, i \in N \mid (s,i) \in \bar{A} \,\}, \quad T_0 := \{t\} \cup \{\, j \in N \mid (j,t) \in \bar{A} \,\}.
\end{aligned}
$$

We apply the well-known algorithm for computing the transitive closure proposed by Warshall in 1962, see e.g., Section 19.3 of Sedgewick [24].

7. Subgradient Method for Lagrangian Dual Problem

For the sake of simplicity we abbreviate $\omega(LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1))$ to $\omega(\boldsymbol{u},\boldsymbol{v})$ in this section. *Lagrangian dual problem*, denoted by (LD), is a problem for finding the smallest upper bound of $\omega(P)$. Namely, it searches for a nonnegative multiplier vector $(\boldsymbol{u},\boldsymbol{v})$ that minimizes $\omega(\boldsymbol{u},\boldsymbol{v})$:

$$(LD) \qquad \left| \begin{array}{ll} \text{minimize} & \omega(\boldsymbol{u},\boldsymbol{v}) \\ \text{subject to} & \boldsymbol{u},\boldsymbol{v} \geq \mathbf{0}. \end{array} \right.$$

The function $\omega(\boldsymbol{u},\boldsymbol{v})$ is piecewise linear convex and not differentiable on the intersection of pieces. One of the most widely used methods for this problem is the *subgradient method*.

Definition 7.1. $(\boldsymbol{g}^u,\boldsymbol{g}^v)$ is said to be a *subgradient* of ω at $(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}}) \geq \mathbf{0}$ when

$$\omega(\bar{\boldsymbol{u}},\bar{\boldsymbol{v}}) + \langle \boldsymbol{g}^u, \boldsymbol{u}-\bar{\boldsymbol{u}}\rangle + \langle \boldsymbol{g}^v, \boldsymbol{v}-\bar{\boldsymbol{v}}\rangle \leq \omega(\boldsymbol{u},\boldsymbol{v})$$

holds for any $(\boldsymbol{u},\boldsymbol{v}) \geq \mathbf{0}$, where $\langle \cdot,\cdot \rangle$ means the inner product.

The following lemma is well known.

Lemma 7.2. *Let $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ denote an optimal solution of the Lagrangian relaxation problem $(LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1))$. Then $(\boldsymbol{g}^u,\boldsymbol{g}^v)$ such that*

$$\begin{aligned} g^u_{ijk} &:= 1 - x(\boldsymbol{u},\boldsymbol{v})_{ij} - x(\boldsymbol{u},\boldsymbol{v})_{jk} + x(\boldsymbol{u},\boldsymbol{v})_{ik} \quad \textit{for } (i,j,k) \in U \\ g^v_{ijk} &:= 0 + x(\boldsymbol{u},\boldsymbol{v})_{ij} + x(\boldsymbol{u},\boldsymbol{v})_{jk} - x(\boldsymbol{u},\boldsymbol{v})_{ik} \quad \textit{for } (i,j,k) \in V \end{aligned}$$

is a subgradient of ω at $(\boldsymbol{u},\boldsymbol{v})$.

We use the following rule to update the multiplier vector $(\boldsymbol{u},\boldsymbol{v})$ to the next iterate $(\boldsymbol{u}^+,\boldsymbol{v}^+)$.

$$u^+_{ijk} := \max\left\{0,\ u_{ijk} - \mu\frac{\omega(\boldsymbol{u},\boldsymbol{v}) - \omega_{\text{low}}}{\|(\boldsymbol{g}^u,\boldsymbol{g}^v)\|^2} g^u_{ijk}\right\} \quad \text{for } (i,j,k) \in U \tag{7.1}$$

$$v^+_{ijk} := \max\left\{0,\ v_{ijk} - \mu\frac{\omega(\boldsymbol{u},\boldsymbol{v}) - \omega_{\text{low}}}{\|(\boldsymbol{g}^u,\boldsymbol{g}^v)\|^2} g^v_{ijk}\right\} \quad \text{for } (i,j,k) \in V, \tag{7.2}$$

where μ is a step size control parameter initially set to 2 and $\|\cdot\|$ is the Euclidean norm. It is known that if ω_{low} in the update formulas is replaced by the optimal value $\omega(P)$, the sequence generated will converge to an optimal solution of the Lagrangian dual problem (LD), see e.g., [12]. However the value $\omega(\boldsymbol{u},\boldsymbol{v})$ does not necessarily decrease when the multiplier vector is updated. We count the number of consecutive failures to decrease the value, and when it amounts to 5, we halve the step size control parameter μ.

When μ falls below 0.005, we increment the constraint index sets U and V and reset μ to its initial value 2. See Section 9 for the details.

8. Heuristics for Good Incumbents

For a given $n \times n$ binary matrix $X := [x_{ij}]_{(i,j)\in N^2}$ let $w_i := \sum_{j\in N} x_{ij} - \sum_{j\in N} x_{ji}$, row-column difference, for each $i \in N$. Ali et al. [1] showed the following lemma and used it in their linear ordering problem formulation.

Lemma 8.1. *Let $X := [x_{ij}]_{(i,j)\in N^2}$ be an $n\times n$ binary matrix with zero diagonal elements. Then $\{w_1,w_2,\ldots,w_n\}$ ranges over $\{n-1,n-3,\ldots,-(n-3),-(n-1)\}$ if and only if X represents a linear ordering.*

When the matrix X satisfies the antisymmetry, i.e., $x_{ji} = 1 - x_{ij}$, the row-column difference w_i could be replaced by the row sum $r_i := \sum_{j \in N} x_{ij}$, which is called the *Copeland score*.

Corollary 8.2. *Suppose that $n \times n$ binary matrix $X := [x_{ij}]_{(i,j) \in N^2}$ satisfies the antisymmetry and has zero diagonal elements. Then the row sum $r_i := \sum_{j \in N} x_{ij}$ ranges over $\{n-1, n-2, \ldots, 1, 0\}$ if and only if X represents a linear ordering.*

Proof. Since $x_{ji} = 1 - x_{ij}$, we see that $w_i = 2r_i - n + 1$ holds. Substituting this for w_i in Lemma 8.1 completes the proof. □

For a solution $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ of $(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1))$, let $\bar{X} := [\bar{x}_{ij}]_{(i,j) \in N^2}$ be the matrix such that

$$\bar{x}_{ij} := \begin{cases} x(\boldsymbol{u}, \boldsymbol{v})_{ij} & \text{for } i < j \\ 0 & \text{for } i = j \\ 1 - x(\boldsymbol{u}, \boldsymbol{v})_{ji} & \text{for } i > j. \end{cases} \tag{8.1}$$

The row sum $r_i := \sum_{j \in N} \bar{x}_{ij}$ of $\bar{X}$ is given by

$$r_i = \sum_{j:j>i} x(\boldsymbol{u}, \boldsymbol{v})_{ij} - \sum_{j:j<i} x(\boldsymbol{u}, \boldsymbol{v})_{ji} + i - 1. \tag{8.2}$$

It is reasonable to think that item with a larger value of r_i should be ranked higher. However, r_i may not accurately reflect the information about which variables are pegged, and it could happen that $r_i < r_j$ even when x_{ij} has been pegged at one or x_{ji} at zero. The descending ordering of the values of r_i may violate the order that we have known is met by every optimal solution. On the other hand, the transitive closure $\bar{B}$ of the arc set

$$B := \{ (i,j) \mid (j,i) \in P_0 \text{ or } (i,j) \in P_1 \}.$$

reflects P_0 and P_1 precisely. Namely, the row-column difference $\bar{w}_i$ of its adjacency matrix satisfies

$$\bar{w}_i > \bar{w}_j \text{ if } (j,i) \in P_0 \text{ or } (i,j) \in P_1.$$

However, lots of row-column differences may fall into a tie before the pegged variables build up.

For a pegged variable x_{ij}, let δ_{ij} be the difference of the positions of item i and j in the optimal linear ordering $\hat{\pi}$, i.e., $\delta_{ij} := |\hat{\pi}^{-1}(i) - \hat{\pi}^{-1}(j)|$. Clearly for $k = 1, 2, \ldots, n-1$, there are $(n-k)$ pairs such that $\delta_{ij} = k$. We consider the variables pegged by the first application of the pegging test. Figure 1 is a scatter plot of

$$\frac{|\{ (i,j) \in N^2_< \mid \delta_{ij} = k, x_{ij} \text{ is pegged by the first application of the pegging test} \}|}{n-k}$$

versus k for $k = 1, 2, \ldots n-1$ for the data `DsumC` of $n = 347$ items. We observed that all pairs with $\delta_{ij} \geq 120$ and 99% of pairs with $\delta_{ij} \geq 42$ were pegged by only the first application of the pegging test. This confirms that $\bar{w}_i$ is credible as the sorting key.

Then we propose to sort the items according to the two keys: $\bar{w}_i$ as the primary key and r_i as the secondary key, which will serve as a tie breaker. The sorting can be done by first sorting according to the secondary key, and then according to the primary key by a *stable* sorting algorithm, e.g., bubble sort. See for example [24].

As heuristics for a good incumbent we first arrange the items as above, and then apply a local search for a further improvement. We observed from some preliminary experiment that 2-opt or 3-opt heuristics is not worth their computational cost, which agrees with the observation reported in Belloni and Lucena [3]. Then we use the following simple heuristic method. Given a linear ordering π, we take an item, say $i = \pi^{-1}(k)$, at the kth position,

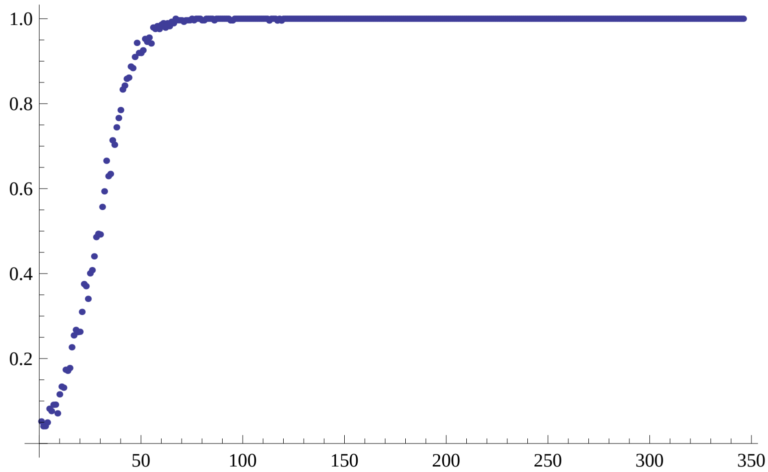

FIGURE 1. Percentage of pegged variables with $\delta_{ij} = k$ vs k

and search for a position in the range of $[\max\{1, k-\beta\}, \min\{n, k+\beta\}]$ such that moving item i to the position improves the objective function value, where β is a fixed positive number. Then we accept it as a temporary incumbent and take the next item $\pi^{-1}(k+1)$ for a possible further improvement.

9. Feasibility Check and Increment of U and V

When μ becomes less than 0.005, we decide that there is no chance of improving the upper bound unless we expand U or V. We add the transitivity constraints violated by the latest optimal solution $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ of $(LR(\boldsymbol{u}, \boldsymbol{v}, P_0, P_1))$. To avoid checking an enormous number of transitivity constraints one by one, we first make the arc set

$$\begin{aligned} A(\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})) : &= \{ (i,j) \in N^2_{\neq} \mid x(\boldsymbol{u}, \boldsymbol{v})_{ij} = 1 \text{ or } x(\boldsymbol{u}, \boldsymbol{v})_{ji} = 0 \} \\ &= \{ (i,j) \in N^2_{\neq} \mid \bar{x}_{ij} = 1 \}, \end{aligned}$$

where $\bar{x}_{ij}$ is defined by (8.1). Then we compute row sum r_i of (8.2), sort the items according to it, and then look for a pair of items such that

$$r_j < r_i \text{ and } (j,i) \in A(\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})),$$

which we call an *upward arc*. Tracing the arcs of $A(\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v}))$ starting from an upward arc, we look for another item, say k, such that the three arcs (j,i), (i,k) and (k,j) form a directed cycle. Clearly this triple violates the transitivity constraint. Furthermore, we see the following lemma.

Lemma 9.1. *The arc set $A(\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v}))$ contains no upward arcs if and only if $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ is a linear ordering.*

Proof. Suppose that $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ is not a linear ordering. Then it violates one of the transitivity constraints. When $x_{ij} + x_{jk} - x_{ik} \leq 1$ is violated, $A(\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v}))$ contains a directed cycle $\{(i,j), (j,k), (k,i)\}$ of length three, and at least one of its arcs form an upward arc. We also see that there is a directed cycle $\{(i,k), (k,j), (j,i)\}$ when $-x_{ij} - x_{jk} + x_{ik} \leq 0$ is violated.

When $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ is a linear ordering, its row sum r_i of $\bar{X}$ ranges over $\{n-1, n-2, \ldots, 1, 0\}$ as in Corollary 8.2. Rearrange the columns and rows simultaneously in the descending order of r_i. Note that the diagonal elements are zero. Clearly the first row consists of a single zero followed by $n-1$ ones, i.e., $(\underline{0},\underbrace{1,1,\dots,1}_{n-1})$. As the induction hypothesis we assume that the hth row is h zeros followed by $n-h$ ones for $h=1,2,\dots,k$. The case of $k=3$ is shown below, where diagonal elements are underlined.

$$\bar{X}=\begin{bmatrix} \underline{0} & 1 & 1 & 1 & \dots & 1 \\ 0 & \underline{0} & 1 & 1 & \dots & 1 \\ 0 & 0 & \underline{0} & 1 & \dots & 1 \\ 0 & 0 & 0 & \underline{0} & & \\ \vdots & \vdots & \vdots & & & \\ 0 & 0 & 0 & & & \end{bmatrix}$$

The $k+1$st row must have $k+1$ zeros and $n-k-1$ ones, and the first k elements are zero by the antisymmetry and the $k+1$st element, which is a diagonal element, is also zero. Therefore it is $k+1$ zeros followed by $n-k-1$ ones, i.e., $(\underbrace{0,0,\dots,0,\underline{0}}_{k+1},\underbrace{1,1,\dots,1}_{n-k-1})$. We see that the matrix $\bar{X}$ is upper triangular, meaning that $A(\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v}))$ has no upward arcs. □

10. Algorithm

The algorithm is composed of the inner and outer cycles. The inner cycle consisting of Step 2 to 7 generates a sequence of Lagrangian multiplier vectors $(\boldsymbol{u},\boldsymbol{v})$, and a sequence of incumbent solutions and values ω_{low}. Some variables are pegged there. The outer cycle expands the constraint index sets U and V.

Step 1 (Initialization)
- (a) Arrange the items according to the row-column difference $\sum_{j\in N} c_{ij}-\sum_{j\in N} c_{ji}$ of cost coefficients and let the linear ordering obtained be the first incumbent solution and let ω_{low} be its objective function value.
- (b) For each consecutive triple (i,j,k) in the incumbent linear ordering, add the transitivity constraints of type 1 and 2 to U and V, respectively.
- (c) $l\leftarrow 0$, $\mu\leftarrow 2.0$, $(\boldsymbol{u},\boldsymbol{v})\leftarrow(\boldsymbol{0},\boldsymbol{0})$.
- (d) $P_0,P_1\leftarrow\emptyset$.
- (e) $\omega_{\text{up}}\leftarrow+\infty$.

Step 2 (Solving $(LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1))$)
- (a) Compute $r(\boldsymbol{u},\boldsymbol{v})_{ij}$ by (4.1).
- (b) Set $x(\boldsymbol{u},\boldsymbol{v})_{ij}$ according to (4.2).
- (c) $\omega_{\text{up}}\leftarrow\min\{\omega_{\text{up}},\omega(LR(\boldsymbol{u},\boldsymbol{v},P_0,P_1))\}$.
- (d) If ω_{up} is not improved, $l\leftarrow l+1$. Otherwise, $l\leftarrow 0$.

Step 3 (Termination)
- (a) If $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ satisfies the optimality condition in Corollary 5.3 with $(\boldsymbol{u},\boldsymbol{v})$, then terminate.
- (b) If $\omega_{\text{up}}-\omega_{\text{low}}<\varepsilon$, then terminate, where ε is a predetermined tolerance to the duality gap.

Step 4 (Heuristics)
- (a) Apply the heuristic method in Section 8 to $\boldsymbol{x}(\boldsymbol{u},\boldsymbol{v})$ for a better solution $\tilde{\boldsymbol{x}}$.
- (b) $\omega_{\text{low}}\leftarrow\max\{\omega_{\text{low}},\text{objective function value of }\tilde{\boldsymbol{x}}\}$.

Step 5 (Pegging Test)
- (a) When $\omega_{\text{up}}-\omega_{\text{low}}<\eta$, then apply the improved pegging test (or the pegging test when $P_0=P_1=\emptyset$).
- (b) Let P_0 and P_1 be the index sets of variables pegged at zero and one, respectively.

Step 6 (Update of μ)
  (a) If $\mu \leq 0.005$, then $\mu \leftarrow 2.0$ and go to Step 8.
  (b) If l reaches 5, $\mu \leftarrow \mu/2$.
Step 7 (Update of $(\boldsymbol{u}, \boldsymbol{v})$)
  (a) Update $(\boldsymbol{u}, \boldsymbol{v})$ according to (7.1) and (7.2).
  (b) Go to Step 2.
Step 8 (Update of U, V)
  (a) Find the transitivity constraints violated by $\boldsymbol{x}(\boldsymbol{u}, \boldsymbol{v})$ and add them to U and V.
  (b) $u_{ijk}, v_{ijk} \leftarrow 0$ for newly added indices (i, j, k).
  (c) Go to Step 2.

11. Computational Results

We coded the algorithm in Java, and run it on a PC with an Intel i3, 3.33 GHz processor and 2 GB of memory. The problem `DsumC` that we solved is a minimum violations ranking problem provided by K. Pedings, College of Charleston. The cost matrix C is based on the point differential matrix of 347 teams in NCAA college basketball for the 2008–2009 season. The problem has 60,031 binary variables and 13,807,130 transitivity constraints. Note that since the cost matrix is an integer matrix, objective function takes an integer value. See Pedings et al. [22] for the details.

Table 1 to 3 show the results in which

- `iteration` shows the number of updates of the Lagrangian multiplier vector $(\boldsymbol{u}, \boldsymbol{v})$,
- `lower bound` shows the incumbent value ω_{low},
- `upper bound` shows the upper bound ω_{up},
- `duality gap` is the difference $\omega_{\text{up}} - \omega_{\text{low}}$,
- $|U| + |V|$ shows the number of transitivity constraints in $U \cup V$,
- $\%^1$ shows the percentage of $|U|+|V|$ to the total number of transitivity constraints,
- $|P_0| + |P_1|$ shows the number of pegged variables,
- $\%^2$ shows the percentage of the pegged variables,
- `time(sec)` shows the computation time in second.

These statistics are given for every 500th iteration.

Table 1 gives the result of the algorithm without pegging tests. After 2212 iterations, the duality gap reduced to less than one, and the incumbent at hand turned out to be an optimal solution. Note that the transitivity constraints being considered account for 0.08%, just a fraction of a percent of the total.

Since the pegging test places a burden on the computation, we did it every 500th iteration. Table 2 gives the result of the algorithm with the ordinary pegging test. It terminated after 2141 iterations in 11.40 seconds, slightly shorter than the computation time when no pegging test was done. Note that about 92% of the variables were eventually pegged.

Table 3 shows the result of the algorithm with the improved pegging test. We applied the ordinary pegging test at the 500th iteration and then the improved pegging test from the 1000th iteration at intervals of 500 iterations. The algorithm found an optimal solution at the 316th iteration, and proved its optimality at the 2193th iteration when the duality gap fell below one. It took the longest computation time due to the burden of the improved pegging test, however, about 95% of the variables were eventually pegged. If we failed to prove the optimality of the incumbent solution by an abortion of computation, this would still provide much information about an optimal solution. We observed that U and V were updated for the first time in the 51th itertion, which led to a sharp decline of the upper bound.

Table 4 shows the result for the problem `DavgC` provided by Pedings based on the same data as `DsumC`. The difference is that the cost matrix consists of fractional cost coefficients. We stopped the computation after 5772 iterations when the duality gap reduced to less than one. The final incumbent may not be optimal, however, more than 95% of variables were pegged.

Table 1. Result for DsumC : no pegging test

iteration	lower bound	upper bound	duality gap	$\lvert U\rvert+\lvert V\rvert$	$\%^1$	$\lvert P_0\rvert+\lvert P_1\rvert$	$\%^2$	time (sec)
1	276183.00	276629.00	446.00	345	0.00	0	0.00	0.01
500	276219.00	276221.38	2.38	10410	0.08	0	0.00	2.35
1000	276219.00	276220.63	1.63	10693	0.08	0	0.00	4.91
1500	276219.00	276220.23	1.23	10865	0.08	0	0.00	7.60
2000	276219.00	276220.04	1.04	11016	0.08	0	0.00	10.32
2212	276219.00	276219.99	0.99	11060	0.08	0	0.00	11.48

Table 2. Result for DsumC : ordinary pegging test

iteration	lower bound	upper bound	duality gap	$\lvert U\rvert+\lvert V\rvert$	$\%^1$	$\lvert P_0\rvert+\lvert P_1\rvert$	$\%^2$	time (sec)
1	276183.00	276629.00	446.00	345	0.00	0	0.00	0.01
500	276219.00	276221.38	2.38	10410	0.08	50836	84.68	2.39
1000	276219.00	276220.63	1.63	10693	0.08	54421	90.65	5.08
1500	276219.00	276220.27	1.27	10916	0.08	55076	91.75	7.87
2000	276219.00	276220.04	1.04	11015	0.08	55386	92.26	10.66
2141	276219.00	276219.99	0.99	11073	0.08	55386	92.26	11.40

Table 3. Result for DsumC : improved pegging test

iteration	lower bound	upper bound	duality gap	$\lvert U\rvert+\lvert V\rvert$	$\%^1$	$\lvert P_0\rvert+\lvert P_1\rvert$	$\%^2$	time (sec)
1	276183.00	276629.00	446.00	345	0.00	0	0.00	0.02
500	276219.00	276221.38	2.38	10410	0.08	50836	84.68	2.48
1000	276219.00	276220.63	1.63	10693	0.08	55953	93.21	8.66
1500	276219.00	276220.30	1.30	10882	0.08	56760	94.55	13.16
2000	276219.00	276220.08	1.08	11046	0.08	57156	95.21	17.36
2193	276219.00	276219.99	0.99	11091	0.08	57156	95.21	18.45

The problems we solved are so limited that more well-organized experiments should be carried out before any conclusion is made.

Table 4. Result for DavgC

iteration	lower bound	upper bound	duality gap	$\lvert U\rvert + \lvert V\rvert$	$\%^1$	$\lvert P_0\rvert + \lvert P_1\rvert$	$\%^2$	time (sec)
1	276269.00	276727.00	458.00	345	0.00	0	0.00	0.02
500	276303.00	276306.05	3.05	11248	0.08	47290	78.78	2.28
1000	276303.00	276305.31	2.31	11548	0.08	53840	89.69	9.27
1500	276303.00	276304.95	1.95	11780	0.09	55361	92.22	14.31
2000	276303.00	276304.77	1.77	11948	0.09	55919	93.15	18.83
2500	276303.00	276304.62	1.62	12025	0.09	55930	93.17	23.22
3000	276303.00	276304.45	1.45	12062	0.09	56384	93.92	27.64
3500	276303.00	276304.35	1.35	12089	0.09	56570	94.23	31.88
4000	276303.00	276304.24	1.24	12138	0.09	56747	94.53	35.94
4500	276303.00	276304.15	1.15	12154	0.09	56896	94.78	39.96
5000	276303.00	276304.09	1.09	12171	0.09	56996	94.94	44.06
5500	276303.00	276304.02	1.02	12187	0.09	57122	95.15	47.88
5772	276303.00	276303.99	0.99	12191	0.09	57122	95.15	49.24

References

[1] I. Ali, W.D. Cook and M. Kress, "On the minimum violation ranking of a tournament", *Management Sciences* **32** (1986) 660–672.

[2] K.J. Arrow, *Social Choice and Individual Values*, John Wiley and Sons, New York, 1951.

[3] A. Belloni and A. Lucena, "Lagrangian heuristics for the linear ordering problem", in: *Metaheuristics: Computer Decision-Making* eds. M.G.C. Resende and J.P. de Sousa, Kluwer, Dordrecht, 2004, pp.37–63.

[4] G. Bolotashvili, M. Kovalev and E. Girlich, "New facets of the linear ordering polytope", *SIAM Journal of Discrete Mathematics* **12** (1999) 326–336.

[5] V. Campos, F. Glover, M. Laguna and R. Martí, "An experimental evaluation of a scatter search for the linear ordering problem", *Journal of Global Optimization* **21** 397–414; DOI: 10.1023/A:1012793906010.

[6] E. Çela, *The Quadratic Assignment Problem: Theory and Algorithms*, Kluwer, Dordrecht, 2010.

[7] I. Charon and O. Hudry, "A branch-and-bound algorithm to solve the linear ordering problem for weighted tournaments", *Discrete Applied Mathematics* **154** (2006) 2097–2116.

[8] I. Charon and O. Hudry, "A survey on the linear ordering problem for weighted or unweighted tournaments", *A Quarterly Journal of Operations Research* **5** (2007) 5–60; DOI: 10.1007/s10288-007-0036-6.

[9] S. Chanas and P. Kobylański, "A new heuristic algorithm solving the linear ordering problem", *Computational Optimization and Applications* **6** (1996) 191–205; DOI: 10.1007/BF00249646.

[10] J.P. Doignon, S. Fiorinia and G. Joret, "Facets of the linear ordering polytope: A unification for the fence family through weighted graphs", *Journal of Mathematical Psychology* **50** (2006) 251–262.

[11] C.G. Garcia, D. Pérez-Brito, V. Campos and R. Martí, "Variable neighborhood search for the linear ordering problem", *Computers and Operations Research* **33** (2006) 3549–3565.

[12] A.M. Geoffrion, "Lagrangian relaxation for integer programming", *Mathematical Programming Study* **2** (1974) 82–114.

[13] S.T. Goddard, "Tournament rankings", *Management Science* **29** (1983) 1385–1392.

[14] C.G. González and D. Pérez-Brito, "A variable neighborhood search for solving the linear ordering problem", *MIC'2001 - 4th Metaheuristics International Conference* pp.181–185.

[15] M. Grötschel, J. Jünger and G. Reinelt, "A cutting plane algorithm for the linear ordering problem", *Operations Research* **32** (1984) 6, 1195–1220.

[16] M. Grötschel, J. Jünger and G. Reinelt, "Facets of the linear ordering polytope" *Mathematical Programming* **33** (1985) 43–60.

[17] J.G. Kemeny and J.L. Snell, *Mathematical Models in the Social Sciences*, Blaisdell, New York, 1972.

[18] J. Leunga, and J. Lee, "More facets from fences for linear ordering and acyclic subgraph polytopes", *Discrete Applied Mathematics* **50** (1994) 185–200.

[19] J.E. Mitchell and B. Borchers, "Solving real-world linear ordering problem using a primal-dual interior point cutting plane method", *Annals of Operations Research* **62** (1996) 253–276; DOI: 10.1007/BF02206819.

[20] J.E. Mitchell and B. Borchers, “Solving linear ordering problems with a combined interior point/simplex cutting plane algorithm”, in: *High Performance Optimization*, H. Frenk et al. eds., pp. 349–366, Kluwer Academic Publishers, 2000.
[21] A. Newman and S. Vempala, “Fences are futile: on relaxations for the linear ordering problem”, in: *Integer Programming and Combinatorial Optimization*, Lecture Notes in Computer Science, 2001, Vol. 2081/2001, 333-347, DOI: $10.1007/3-540-45535-3_26$.
[22] K. Pedings, A.N. Langville and Y. Yamamoto, “A minimum violations ranking method”, to appear in *Optimization and Engineering*.
[23] G. Reinelt, *The Linear Ordering Problem: Algorithms and Applications*, Helderman, Berlin, 1985.
[24] R. Sedgewick, *Algorithms in C*, 3rd Edition, Addison-Wesley, Boston, 2002.

Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba, Ibaraki 305-8573, Japan
E-mail address: {sukega60, yamamoto}@sk.tsukuba.ac.jp, zhangliyuan120@gmail.com