Matchings in Colored Bipartite Networks

Tongnyoul Yi[c], Katta G. Murty[+a], Cosimo Spera[b]
[a]Dept. of IOE, University of Michigan, Ann Arbor, MI 48109-2117, USA
[b]Dept. of QM, University of Siena, Piazza S. Francesco, 53100 Siena, Italy
[c]Samsung Data Systems, Seoul, S. Korea 120-020

August 13, 2000

Abstract

In $K(n, n)$ with edges colored either red or blue, we show that the problem of finding a *solution matching*, a perfect matching consisting of exactly r red edges, and $(n - r)$ blue edges for specified $0 \leq r \leq n$, is a nontrivial integer program. We present an alternative, logically simpler proof of a theorem in [3] which establishes necessary and sufficient conditions for the existance of a solution matching and a new $O(n^{2.5})$ algorithm. This shows that the problem of finding an assignment of specified cost r in an assignment problem on the complete bipartite graph with a $0-1$ cost matrix is efficiently solvable.

Key words assignment problem, $0-1$ cost matrix, extreme point with specified objective value.

[+]Author for correspondence

1 Introduction

A problem of interest in core management of pressurized water nuclear reactors is [2]: given an $n \times n$ cost matrix $c = (c_{ij})$ and the desired objective value r, find $x = (x_{ij})$ satisfying

$$\left.\begin{array}{l} \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1 \text{ to } n \\ \\ \sum_{i=1}^{n} x_{ij} = 1, \quad j = 1 \text{ to } n-1 \end{array}\right\} \tag{1}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} = r \tag{2}$$

$$x_{ij} = \quad 0 \text{ or } 1 \text{ for all } i, j \tag{3}$$

We assume that the rank of the coefficient matrix of (1), (2) is $2n$. Problem (1)−(3) has been shown to be NP-hard in [1]. Papadimitriou [6] posed the question whether (1)−(3) can be solved efficiently when c is a 0−1 matrix, calling it a *mysterious problem*. This is the problem we consider in this paper, i.e., where c is a $0-1$ matrix.

Karzanov in [4] studied the problem in general graphs and derived necessary and sufficient conditions for the existence (or nonexistence) of a solution for this special problem (Theorem 3 in the following). A solution algorithm, although not given, may be derived from the proof in [4], and its polynomiality is quite transparent. In this paper we provide a simpler proof of these conditions using an analysis based on 2×2 subgraphs, and a new $O(n^{2.5})$ algorithm, which arises from these conditions.

In general, this problem is stated on an incomplete bipartite graph, i.e., we are given a subset $F \subset \{1, \ldots, n\} \times \{1, \ldots, n\}$ and are required to also satisfy the additional conditions $x_{ij} = 0$ for all $(i, j) \in F$. This problem on the incomplete bipartite graph is perhaps harder, so far no efficient algorithm is known for finding a solution matching in an incomplete bipartite graph. Karzanov [4] considered only the complete bipartite graph case, and we will do the same.

2 Some Preliminaries

Let $G = K(n, n)$, the $n \times n$ complete bipartite graph. Associate the variable x_{ij} in (1) with the edge (i, j) in G. In the sequel c will always be a 0−1 matrix, and r will be an integer satisfying $0 \leq r \leq n$. Color the edge (i, j) in G blue if $c_{ij} = 0$, red if $c_{ij} = 1$. G_R, G_B denote the subgraphs with red and blue edges respectively. With this representation, (1)−(3) is the following problem.

PROBLEM 1 Input: G_R, G_B, the partition of G into the red and blue subgraphs, and the *requirement vector* $[r, n-r]$ where $0 \leq r \leq n$. Output needed: A *solution matching* which is a perfect matching in G with exactly r red and $n - r$ blue edges.

The following lemma, whose proof is easily obtained by standard arguments [4], shows that this problem is nontrivial.

LEMMA 1 *If c is a 0–1 matrix, the determinant of a basis for (1), (2) may not be ± 1, but it is always between $-(n+1)$ to $+(n+1)$.*

As an example, when $n = 2$ and $c = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ the graph is in Figure 2. System (1), (2) for this example is a square nonsingular system of equations with the determinant of the coefficient matrix equal to 2. For $r = 0$, 2, the solution of this system is integral, but for $r = 1$ its only solution is $(x_{11}, x_{12}, x_{21}, x_{22})^T = (1/2, 1/2, 1/2, 1/2)^T$. So there is no solution matching when $r = 1$ in this example.

So, the LP relaxation of (1)–(3) may have basic feasible solutions which are not integral, and hence solving (1)–(3) is a nontrivial integer program.

3 Some Procedures

Let M_R be a matching of cardinality r in G_R, and M_B a matching of cardinality $n - r$ in G_B. With respect to $M_r \cup M_B$ a node in G is said to be a *good node* if it has exactly one edge of $M_R \cup M_B$ incident at it, *exposed node* if it has no edges of $M_R \cup M_B$ incident at it, and a *bad node* if it has both a red and a blue edge of $M_R \cup M_B$ incident at it. Now we try to convert all the bad and exposed nodes into good nodes using procedures 1, 2, 3 described below, while keeping the cardinalities of the red and blue matchings at r, $n - r$ respectively throughout.

PROCEDURE 1: To convert a bad and exposed node together into good nodes.

Let i_1 be any bad node with $(i_1, j_1), (i_1, j_2)$ as the red, blue matching edges incident at it, see the left side of Figure 1 (in the figures dashed edges are matching edges, solid edges are nonmatching edges). There must be an exposed node, i_0, in the same set of the bipartition for G as i_1. Since G is complete, both the edges (i_0, j_1) and (i_0, j_2) exist. If (i_0, j_1) is red, rematch the red alternating path $\mathcal{P}_R$: $i_1, (i_1, j_1), j_1, (i_0, j_1), i_0$ (i.e., make (i_1, j_1) into a nonmatching edge and (i_0, j_1) into a matching edge). This converts both i_1 and i_0 into good nodes (see the right side of Figure 1).

Similarly, if (i_0, j_1) is blue, and (i_0, j_2) is also blue, rematch the blue alternating path $\mathcal{P}_B$: $i_1, (i_1, j_2), j_2, (i_0, j_2), i_0$; this now converts i_1, i_0 into good nodes.

If (i_0, j_1) is blue and (i_0, j_2) is red, this procedure is unable to convert the pair (bad node i_1, exposed node i_0) into good nodes (see Figure 2).

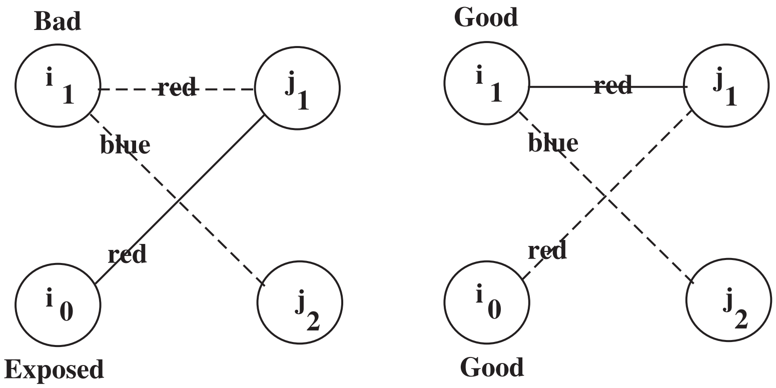

Figure 1: A bad, exposed node pair joined by a red alternating path on left (dashed edges are matching edges, solid edges are nonmatching edges); on right same graph after rematching this path.

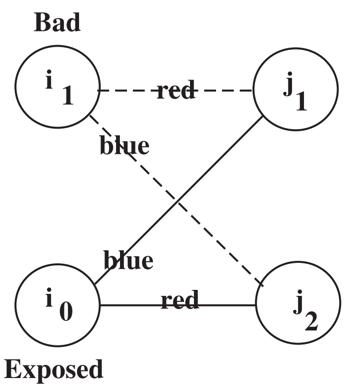

Figure 2: Failure of Procedure 1 to convert the bad exposed node pair. A 2×2 irreducible subgraph.

PROCEDURE 2: To convert two bad nodes not joined by a matching edge, into good nodes.

Let i_1, p_1 be two bad nodes in the current union $M_R \cup M_B$ which could not be converted into good nodes by applying Procedure 1 with any exposed nodes. Here we consider the case where either i_1, p_1 both belong to the same set in the bipartition for G (so there is no edge joining i_1 and p_1 in G), or they belong to different sets in the bipartition for G but (i_1, p_1) is not a matching edge. Let $(i_1, j_1), (p_1, q_1)$ be the red matching edges; and $(i_1, j_2), (p_1, q_2)$ the blue matching edges incident at them. There exist distinct exposed nodes i_0 p_0 in the same set of the bipartition for G as i_1 p_1. Subgraphs induced by $\{i_1, j_1, j_2, i_0\}$, $\{p_1, q_1, q_2, p_0\}$ are as in Figure 3 since Procedure 1 failed to convert either of the pairs $\{i_1, i_0\}$, $\{p_1, p_0\}$ into a good pair. Make $(i_1, j_1), (p_1, q_2)$ into nonmatching edges, and $(i_0, j_1), (p_0, q_2)$ into matching edges. This converts i_1, i_0, p_1, p_0 into good nodes; terminate the procedure.

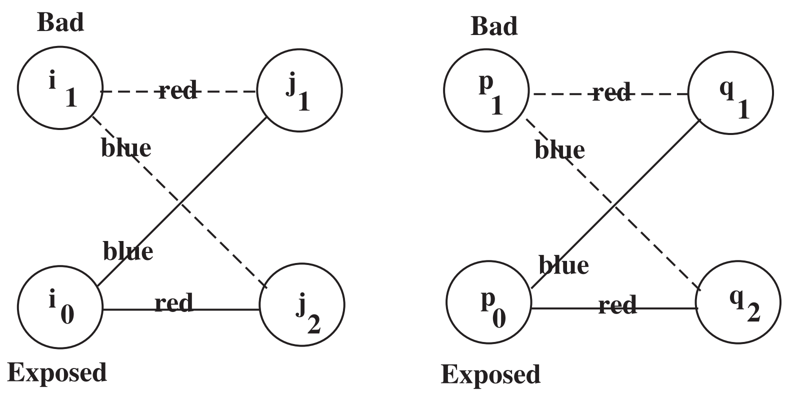

Figure 3: Subgraph induced by $\{i_1, j_1, j_2, i_0\}$ on the left, and that induced by $\{p_1, q_1, q_2, p_0\}$ on the right.

PROCEDURE 3: To convert two bad nodes joined by a matching edge into good nodes.

Let i_1 be a bad node with red and blue matching edges $(i_1, j_1), (i_1, j_2)$ incident at it. Suppose j_1 is also a bad node with (i_2, j_1) as the blue matching edge incident at it, but j_2 is a good node (Figure 4). There must be an exposed node, i_0, in the set of the bipartition for G as i_1. If (i_0, j_2) is blue, Procedure 1 applies. If (i_0, j_2) is red, make (i_1, j_1) into a nonmatching edge, and (i_0, j_2) into a matching edge. See Figure 5. This change converts j_2 into a bad node, but i_1, j_1 both become good nodes, thus reducing the number of bad nodes by one.

Apply Procedure 1 as often as possible, or Procedures 2 or 3 as appropriate taking the bad nodes in pairs, reducing the number of bad nodes to either 0 (leading to a solution matching), or 1. In the latter case there must be exactly one exposed node in the same set of the bipartition for G as the bad node; and at this stage the bad node, its mates, and the exposed node form a 2×2 subgraph as in Figure 2, while all the remaining nodes are well matched by node disjoint matching edges. In this case, this 2×2 subgraph in Figure 2 is called the *2×2 irreducible subgraph* at this stage.

4 Algorithm for a Special case

A 2×2 subgraph of G is said to be a *2×2 odd subgraph* if it contains either 1 red and 3 blue edges, or 1 blue and 3 red edges.

We will find it convenient to associate edges in G with cells in a two dimensional $n \times n$ array as is usually done in discussions of the assignment problem in operations

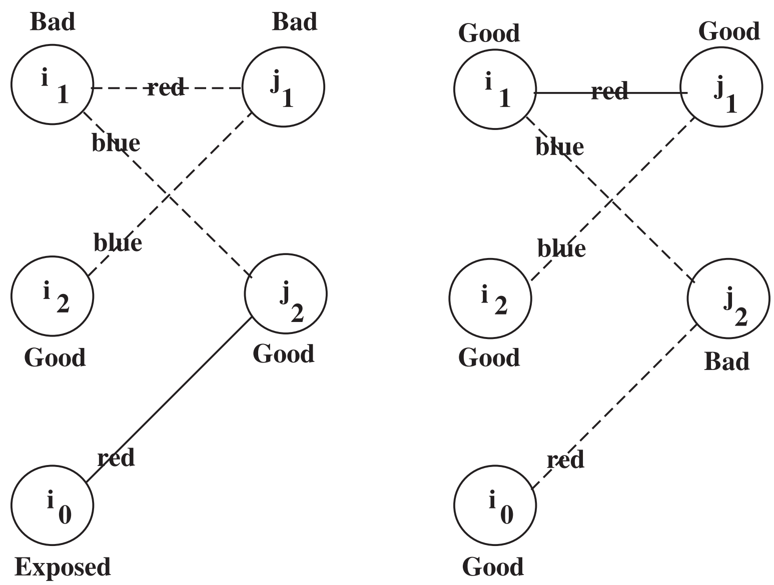

Figure 4: On the left is the subgraph on which Procedure 3 applies. The right shows the same subgraph after applying Procedure 3.

research literature (for example [5]). The cell (i, j) in the array associated with the edge (i, j) in G is colored with the same color as the edge. For the sake of clarity, let $(A = \{A_1, \ldots, A_n\}, B = \{B_1, \ldots, B_n\})$ be the bipartition for G; i.e., a general edge in G is (A_i, B_j) for $i, j = 1$ to n. With this definition, we will denote G by (A, B).

LEMMA 2 *The $n \times n$ complete bipartite graph $G = (A, B)$ has no 2×2 odd subgraph if and only if there exist partitions $A = A' \cup A''$, $B = B' \cup B''$, such that all the edges in $(A', B') \cup (A'', B'')$ have the same color, and all the edges in $(A', B'') \cup (A'', B')$ have the other color.*

Proof. If partitions exist as stated in the lemma, it is easy to verify that no 2×2 odd subgraph exists. To show the converse, suppose G has no 2×2 odd subgraph.

Let $K_1, \ldots K_t$ be the components of G_R, $t \geq 1$. For $v = 1$ to t let $I_v \subset A, J_v \subset B$ be the subsets of nodes on edges in K_v. The following results clearly imply the lemma.

(i) Each K_v must be complete bipartite. For if not, K_v contains red edges $(u, p), (w, p), (w, q)$ while the edge (u, q) is blue, so the 2×2 subgraph induced by $\{u, w, p, q\}$ is odd.

(ii) $t \leq 2$. For if $t \geq 3$, select $u \in I_1, p \in J_1, w \in I_2, q \in J_3$. Then (u, p) is red, while $(u, q), (w, p), (w, q)$ are blue giving a 2×2 odd subgraph induced by $\{u, w, p, q\}$.

(iii) If $t = 1$ at least one of $|I_1|, |J_1|$ is n. For otherwise the 2×2 subgraph induced by $\{u, w, p, q\}$ where $u \in I_1, p \in J_1, w \in A \backslash I_1, q \in B \backslash J_1$ is odd.

(iv) If $t = 2$, then $I_1 \cup I_2 = A$, $J_1 \cup J_2 = B$. For if $I_1 \cup I_2 \neq A$, then the subgraph induced by $\{u, w, p, q\}$ where $u \in I_1, p \in J_1, q \in J_2, w \in A \backslash (I_1 \cup I_2)$ is odd. ∎

5 Algorithm when G has no 2×2 odd subgraph

Whether G satisfies the hypothesis in Lemma 2 can be checked in $O(n^2)$ time. If it does, by rearranging the rows [columns] corresponding to nodes in the sets A', A'' $[B', B'']$ together, the two dimensional array representation of G is as in Figure 5 with the cells in the blocks D_1, D_3 red, and those in D_2, D_4 blue.

	B'	B''
A'	D_1 Red	D_2 Blue
A''	D_4 Blue	D_3 Red

Figure 5: Partition of G when it has no 2×2 odd subgraph.

Lemma 3 *Let A', A'', B', B'', D_1 to D_4 be as in Lemma 2 or Figure 5. For $t = 1$ to 4 let r_t be the number of matching edges from block D_t in a solution matching for requirement vector $[r, n-r]$. Then*

$$\left.\begin{array}{ll} r_1 = (-n + r + |A'| + |B'|)/2, & r_2 = (n - r + |A'| - |B'|)/2 \\ r_3 = (n + r - |A'| - |B'|)/2, & r_4 = (n - r - |A'| + |B'|)/2 \end{array}\right\} \tag{4}$$

Proof. From (1), (2), (3), we see that $r_1 + r_2 = |A'|$, $r_1 + r_4 = |B'|$, $r_2 + r_3 = |B''| = n - |B'|$, $r_1 + r_3 = r$. This system of four equations in r_1, r_2, r_3, r_4 has the unique solution given in (4). ∎

Theorem 1. *Let A', B', A'', B'' be as in Lemma 2. A solution matching for the requirement vector $[r, n-r]$ exists in G iff $n + r + |A'| + |B'|$ is even, and r_1 to r_4 in (4) are nonnegative.*

Proof. If a solution matching exists, define r_1 to r_4 as in Lemma 3, and verify that these quantities given by (4) are integers only if $n + r + |A'| + |B'|$ is even.

If $n + r + |A'| + |B'|$ is even, and $(r_1, \ldots, r_4)$ given by (4) are all ≥ 0, the solution to the four equations in the proof of Lemma 3, is nonnegative and integral. Let $P_1 \subset A'$ with $|P_1| = r_1$, $P_2 = A' \backslash P_1$; $P_3 \subset A''$ with $|P_3| = r_3$, $P_4 = A'' \backslash P_3$; $Q_1 \subset B'$ with $|Q_1| = r_1$, $Q_4 = B' \backslash Q_1$; $Q_2 \subset B''$ with $|Q_2| = r_2$, $Q_3 = B'' \backslash Q_2$. Then for $t = 1$ to 4, (P_t, Q_t) is $K(r_t, r_t)$, let M_t be a perfect matching in (P_t, Q_t). Then $\cup_{t=1}^{4} M_t$ is a solution matching in G for the requirement vector $[r, n-r]$. ∎

Theorem 2 *Let A', A'', B', B'' be as in Lemma 2. Consider the array representation of G as in Figure 5. For $t = 1$ to n let $a_t = \{(A_i, B_{i+t-1}) : i = 1$ to $n - t + 1$, $(A_i, B_{i-n+t-1}) : i = n - t + 2$ to $n\}$ be the perfect matching represented by the tth diagonal in Figure 5. Then a solution matching for the requirement vector $[r, n - r]$ exists in G iff one of these diagonal perfect matchings $a_t, t = 1$ to n has r red and $n - r$ blue edges.*

Proof. The "if" part is obvious. Conversely if a solution matching exists, the rows and columns in the array can be rearranged so that the cells in this solution matching are along one of the diagonal positions in the array, implying the result. ∎

Theorem 2 implies that if G has no 2×2 odd subnetwork, then all numbers r for which solution matchings exist for the requirement vector $[r, n - r]$ in G have the same odd-even parity and form an arithmetic progression in which consecutive elements differ by 2.

6 Conditions for the Nonexistence of a Solution Matchings

Theorem 3 *If G_R has a matching of cardinality r and G_B has a matching of cardinality $n - r$, where $n \geq 4$ and $2 \leq r \leq n - 2$ and there is no solution matching in G for the requirement vector $[r, n - r]$, then G has no 2×2 odd subgraph.*

Proof. The theorem is easily verified for $n = 4$. So, assume $n \geq 5$ and set up an induction hypothesis that the theorem is true for graphs of order $(n - 1) \times (n - 1)$.

Assuming the hypothesis and the results in Section 2 imply that there exists matchings $\hat{M}_R, \hat{M}_B$ in G_R, G_B satisfying $|\hat{M}_R| = r, |\hat{M}_B| = n - r$, and $\hat{M}_R \cup \hat{M}_B$ contains only one bad node. This is guaranteed by repeatedly applying Procedures 1, 2, 3 as in Section 2.

Without any loss of generality assume that $r \geq n - r$. Let (A_n, B_n) be a matching edge in $\hat{M}_R$, with A_n, B_n being good nodes. Let $\bar{A} = A \backslash \{A_n\}, \bar{B} = B \backslash \{B_n\}, \bar{G} = (\bar{A}, \bar{B})$. Since G has no solution matching for the requirement vector $[r, n-r]$, $\bar{G}$ has no solution matching for the requirement vector $[r-1, n-r]$. And $\hat{\bar{M}}_R = \hat{M}_R \backslash \{(A_n, B_n)\}, \hat{\bar{M}}_B = \hat{M}_B$ are red and blue matchings in $\bar{G}$ of cardinalities $r - 1, n - r$. Hence by the induction hypothesis and Lemma 2, there exist partitions $\bar{A} = \bar{A}' \cup \bar{A}'', \bar{B} = \bar{B}' \cup \bar{B}''$ such that all the edges in $(\bar{A}', \bar{B}') \cup (\bar{A}'', \bar{B}'')$ are red, and those in $(\bar{A}', \bar{B}'') \cup (\bar{A}'', \bar{B}')$ are all blue (Figure 6).

	B'	B''	B_n
A'	Red	Blue	
A''	Blue	Red	
A_n			Red

Figure 6: Partition of G.

The bad node in $\hat{\bar{M}}_R \cup \hat{\bar{M}}_B$, its two mates and the exposed node in $\bar{G}$ define a 2×2 irreducible subgraph, let it be $E = (\{A_p, A_q\}, \{B_\ell, B_m\})$. The edges in $\hat{\bar{M}}_R \cup \hat{\bar{M}}_B$ outside of E are a pairwise node disjoint set of edges with $r - 2$ red and $n - r - 1$ blue edges. Also, from the structure in Figure 6 it can be verified that given any $A_{p_1} \in \bar{A}', A_{q_1} \in \bar{A}'', B_{\ell_1} \in \bar{B}', B_{m_1} \in \bar{B}''$; matching changes inside $\bar{G}_R, \bar{G}_B$ can be made so that in the resulting union $(\{A_{p_1}, A_{q_1}\}, \{B_{\ell_1}, B_{m_1}\})$ is the irreducible subgraph.

If the 3×3 subgraph $H = (\{A_p, A_q, A_n\}, \{B_\ell, B_m, B_n\})$ has a solution matching for the requirement vector [2, 1]; then by combining it with the matching edges in $\hat{\bar{M}}_R \cup \hat{\bar{M}}_B$ outside of E, we get a solution matching in G for the requirement vector $[r, n-r]$. Hence there exists no solution matching in H for the requirement vector [2, 1]. This implies that cells $(A_p, B_n), (A_n, B_\ell)$ have the same color; cells $(A_q, B_n), (A_n, B_m)$ have the same color; and that the cells $(A_p, B_n), (A_q, B_n), (A_n, B_\ell), (A_n, B_m)$ cannot all be red. By varying A_p in $\bar{A}'$, A_q in $\bar{A}''$, B_ℓ in $\bar{B}'$, B_m in $\bar{B}''$, we conclude that all cells in $(\bar{A}', B_n) \cup (A_n, \bar{B}')$ have the same color, say color 1; and that all cells in $(\bar{A}'', B_n) \cup (A_n, \bar{B}'')$ have the same color, say color t; and that it is not possible for both color 1 and color t to be red.

Suppose color 1 and color t are both blue. In this case the array representation for G has the color pattern in Figure 6 with all cells in the blank spaces in the row of A_n and the column of B_n being blue. Let $\mathcal{M}_1$ be the set of all perfect matchings in G with the red edge (A_n, B_n) as a matching edge. For each $1 \leq i \leq n-1$, $1 \leq j \leq n-1$, let $\mathcal{M}_{2ij}$ be the set of all perfect matchings in G with blue edges $(A_n, B_j), (A_i, B_n)$ as matching edges.

When edge (A_n, B_n) is deleted from each matching in $\mathcal{M}_1$ we get the set $\bar{\mathcal{M}}_1$ of perfect matchings in the array of order $(n-1) \times (n-1)$ obtained by deleting the row of A_n and the column of B_n from the array in Figure 6. Similarly $\bar{\mathcal{M}}_{2ij}$, the set of matchings obtained by deleting cells $(A_n, B_j), (A_i, B_n)$ from each matching in $\mathcal{M}_{2ij}$, is the set of perfect matchings in array of order $(n-2) \times (n-2)$ obtained by deleting the rows of A_n, A_i and the columns of B_j, B_n from the array in Figure 6. This array of order $(n-1) \times (n-1)$, and each of the arrays of order $(n-2) \times (n-2)$ belong to the special case discussed in Section 4, and hence the set of values that the number of red cells can take among matchings in $\bar{\mathcal{M}}_1, \bar{\mathcal{M}}_{2ij}$ is characterized by the results in Lemma 3, and Theorems 1, 2. From this it can be verified that G has a perfect matching containing exactly r red cells, contradicting the hypothesis. This implies that color 1 and color t cannot both be blue.

So color 1 and color t have to be different. Now define $A' = \bar{A}', B' = \bar{B}', A'' = \bar{A}'' \cup \{A_n\}, B'' = \bar{B}'' \cup \{B_n\}$ if color t is red; otherwise define $A' = \bar{A}' \cup \{A_n\}, B' = \bar{B}' \cup \{B_n\}, A'' = \bar{A}'', B'' = \bar{B}''$ if color 1 is red. Then all edges in $(A', B') \cup (A'', B'')$ are red and all edges in $(A', B'') \cup (A'', B')$ are blue. Hence by Lemma 2, G has no 2×2 odd subgraph, establishing the statement in the theorem for G. Hence by induction, the theorem holds in general. ∎

Definition 1: Suppose $G = (A, B)$, has representation as in Figure 5. There is a horizontal and a vertical line in the array in Figure 5 seperating the colors. These are the thick lines in Figure 5 when all the sets A', A'', B', B'' are nonempty. If B'' [A''] $= \emptyset$, we define the vertical [horizontal] line to be the rightmost vertical [bottommost horizontal] boundary line of the array. The point of intersection of these horizontal and vertical lines is called the *crossover point*, in this array representation. The *crossover cells* in this array representation are defined to be the cells in the array that contain the crossover point as either their upper right corner point or lower left corner point. Thus, when both A'', B'' are nonempty, there are two crossover cells; when one of A'', B'' is empty and the other is not, there is one crossover cell; and when both A'', B'' are empty there is no crossover cell.

Lemma 4 *Let $G = (A, B)$ have representation as in Figure 5. G has a solution matching for the requirement vector $[n-1, 1]$ iff in this array representation, one of its main diagonal cells is a crossover cell.*

Proof. If a crossover cell is on the main diagonal, then the cells along the main diagonal form a solution matching for the requirement vector $[n-1, 1]$. If there is no crossover cells, or when they exist but none of them is on the main diagonal, there is no t such that the tth diagonal in the array contains exactly one blue cell, and by the results in Section 4, there is no solution matching for the requirement vector $[n-1, 1]$. ∎

Lemma 5 *If there is no 2×2 odd subgraph with 1 blue and 3 red edges in G, then there exist partitions $A = A^1 \cup \ldots \cup A^k$, $B = B^1 \cup \ldots \cup B^k$, $k > 2$ such that $A^t \neq \emptyset, B^t \neq \emptyset$ for all t = 1 to $k-1$, and all edges in (A^t, B^t) are red for t = 1 to k' where k' is either $k-1$ or k, and all the other edges in G are blue (Figure 7).*

Proof. Clearly the lemma holds for G when $n = 2$. Set up an induction hypothesis that the lemma holds for complete bipartite graphs of order $(n-1) \times (n-1)$.

If there is no red edge in G, the lemma holds for G with $k = 1$, and $k' = k - 1$. Otherwise select a red edge, (A_p, B_q). Let $\bar{A} = A \backslash \{A_p\}$, $\bar{B} = B \backslash \{B_q\}$. Since G has no 2×2 odd subgraph with 1 blue and 3 red edges, $\bar{G} = (\bar{A}, \bar{B})$ does not either. So, by the induction hypothesis, there exist partitions $\bar{A} = \bar{A}^1 \cup \ldots \cup \bar{A}^{k_1}$, $\bar{B} = \bar{B}^1 \cup \ldots \cup \bar{B}^{k_1}$, such that $\bar{A}^t, \bar{B}^t$ are both nonempty for all $t = 1$ to $k_1 - 1$, and all the edges in $(\bar{A}^t, \bar{B}^t)$ are red for $t = 1$ to k_1' where k_1' is either $k_1 - 1$ or k_1, and all the other edges in $\bar{G} = (\bar{A}, \bar{B})$ are blue.

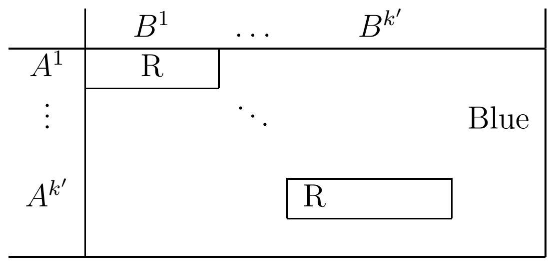

Figure 7: Partition of G.

Case 1 B_q has another red edge other than (A_p, B_q) incident at it: Suppose (A_h, B_q) is red with $h \neq p$. Let $\bar{A}^b$ be the set in the partition of $\bar{A}$ that contains A_h. The facts here, imply that (A_p, B_j) is red for all $B_j \in \bar{B}^b$, and blue for all $B_j \notin \bar{B}^b$, and similarly that (A_i, B_q) is red for all $A_i \in \bar{A}^b$ and blue for all $A_i \notin \bar{A}^b$. Hence, if we define $A^t = \bar{A}^t$ for all $t \neq b$, and $A^b = \bar{A}^b \cup \{A_p\}, B^b = \bar{B}^b \cup \{B_q\}$, then the partitions $A = A^1 \cup \ldots \cup A^{k_1}, B = B^1 \cup \ldots \cup B^{k_1}$ satisfy the conditions in the lemma. Hence the lemma holds for G.

Case 2 (A_p, B_q) is the only red edge in G incident at B_q: So, in this case (A_i, B_q) is blue for all $A_i \neq A_p$. The facts here imply that (A_p, B_j) is blue for all $B_j \in \bar{B}^1 \cup \ldots \cup \bar{B}^{k_1'}$. Now define $k = 1 + k_1$, $A^1 = \{A_p\}$, $B^1 = \{B_q\} \cup \{B_j : B_j \notin \bar{B}^1 \cup \ldots \cup \bar{B}^{k_1'}$ and (A_p, B_j) is red$\}$; $A^{t+1} = \bar{A}^t$, for $t = 1$ to k_1; $B^{t+1} = \bar{B}^t$ for $t = 1$ to k_1'; and when $k_1' = k_1 - 1$, $B^k = \bar{B}^{k_1} \backslash B^1$; $k' = 1 + k_1'$. Then $A = A^1 \cup \ldots \cup A^k, B = B^1 \cup \ldots \cup B^k$ are partitions of A, B satisfying the conditions in the lemma, hence the result in the lemma is true for G.

Thus the lemma holds for the $n \times n$ bipartite graph G under the induction hypothesis, and by induction, it holds for all n. ∎

Definition 2: Suppose $G = (A, B)$ has representation as in Figure 7. There are $k - 1$ horizontal lines ($k - 1$ vertical lines) separating the subsets in the row (column) partition isolating the red blocks. In case A^k (or B^k) is empty, the bottommost horizontal line (rightmost vertical line) is considered to be one of the horizontal (vertical) lines. For $t = 1$ to $k - 1$, the point of intersection of the tth horizontal line from the top, and the tth vertical line from the left, is called the *tth crossover point* in this array representation. For $t = 1$ to $k - 1$, the cells containing the tth crossover point either as their upper right corner point or as their lower left corner point are called the *tth crossover cells* in this array representation.

Theorem 4. *Let $n \geq 3$, and G be as in Figure 7. Suppose $k' = k$, and none of the crossover cells as specified in Definition 2 are on the main diagonal. Then G has no perfect matching with 1 blue and $(n - 1)$ red edges.*

Proof. It can be verified that the theorem holds when $n = 3$. Let $n \geq 4$. Set up an induction hypothesis that the theorem holds for complete bipartite graphs of order $(n-1) \times (n-1)$ that have array representation as in Figure 7 with $k' = k$.

Select a red edge in G, say (A_i, B_j). When the row corresponding to A_i, and the column corresponding to B_j are deleted from the array representation of G, what remains is the array representation of $\bar{G} = (A \backslash \{A_i\}, B \backslash \{B_j\})$. $\bar{G}$ either has an array representation as in Figure 7 with $k' = k$, or as in Figure 5, and has a crossover cell along its main diagonal iff G has one or more crossover cells along its main diagonal, and (A_i, B_j) is a main diagonal cell of G whose deletion leaves at least one of the crossover cells along the main diagonal of G in $\bar{G}$. Since G has no crossover cells along its main diagonal, $\bar{G}$ satisfies the same property. Hence by Lemma 4 and the induction hypothesis, $\bar{G}$ has no perfect matching with 1 blue and $(n-2)$ red cells, so there is no perfect matching in G with 1 blue and $(n-1)$ red cells containing (A_i, B_j) as a matching edge. A similar argument shows that the same statement is true for every red edge (A_i, B_j) in G, i.e., G has no perfect matching with 1 blue and $(n-1)$ red edges. So the theorem holds for the $n \times n$ complete bipartite graph G under the induction hypothesis, and by induction it holds for all $n \geq 3$. ∎

Theorem 5. *G has no 2×2 odd subgraph with 1 blue and 3 red edges iff each connected component of G_R is a complete bipartite subgraph. This condition can be checked, and a 2×2 odd subgraph of G with 1 blue and 3 red edges can be found if it is violated, with $O(n^2)$ effort.*

Proof. If G has a 2×2 odd subgraph with 1 blue and 3 red edges, the connected component of G_R containing the nodes on this subgraph is not complete bipartite. Combining this fact with the result in Lemma 5, we conclude that G has no 2×2 odd subgraph with 1 blue and 3 red edges iff each connected component of G_R is a complete bipartite subgraph.

When this condition is violated, let K_1 be a connected component of G_R which is not complete bipartite. So, you can find nodes i_1, j_1 in K_1 contained in different sets of the bipartition of G, such that (i_1, j_1) is not an edge in K_1, i.e., (i_1, j_1) is a blue edge in G.

With the length of each edgs equal to 1, find a shortest simple path from i_1 to j_1 in K_1. Suppose it is $\mathcal{P}$,

$$i_1 = u_0, (i_1 = u_0, v_1), v_1, (v_1, u_1), u_1, \ldots, (u_{s-1}, v_s), v_s, (v_s, u_s), u_s, (u_s, j_1), j_1$$

where $s \geq 1$ since (i_1, j_1) is not an edge in K_1. Also, since $\mathcal{P}$ is a shortest path from i_1 to j_1 in K_1, (u_{s-1}, j_1) is not an edge in K_1 even though u_{s-1}, j_1 are in different sets of the bipartition of G, i.e., (u_{s-1}, j_1) is a blue edge in G. Hence the 2×2 subgraph of G induced by $\{u_{s-1}, v_s, u_s, j_1\}$ has 1 blue and 3 red edges.

Each of the operations involved in this work (like finding the connected components of G_R; checking whether each of these connected components is complete bipartite; finding a blue edge in G joining two nodes i_1, j_1 in a connected component of G_R that is not complete bipartite; finding a shortest path in that connected component) can be carried out with $O(n^2)$ effort. So, the existence of a 2×2 odd subgraph with 1 blue and 3 red edges can be checked, and one of them found if they exist, with $O(n^2)$ effort. ∎

Theorem 6. *Let t be the number of connected components in G_R. G has no 2×2 odd subgraph (either with 1 red and 3 blue edges, or with 1 blue and 3 red edges) iff the following conditions hold.*

(i) *t must be ≤ 2, and each of the connected components of G_R must be a complete bipartite subgraph of G.*

(ii) *If G_R is connected (i.e., $t = 1$), then it must contain all the nodes in at least one of the two sets in the bipartition of G. If $t = 2$, both the connected components of G_R put together must contain all the nodes in G.*

With at most $O(n^2)$ effort we can check whether these conditions are satisfied; if they are, find the partitions $A = A' \cup A'', B = B' \cup B''$ as described in Lemma 2; and if they are not, find a 2×2 odd subgraph of G.

Proof. This result follows directly from the proof of Lemma 2.

The main work in checking whether these conditions are satisfied is to find the connected components of G_R, and check whether each of these connected components is a complete bipartite subgraph of G. Each of these can be carried out with $O(n^2)$ effort, so the overall effort needed is $O(n^2)$.

If both conditions (i), (ii) are satisfied and $t = 2$, let the two connected components of G_R be $(A', B', A' \times B'), (A'', B'', A'' \times B'')$. In this case all the sets A', A'', B', B'' are nonempty, and by (ii), $A = A' \cup A'', B = B' \cup B''$; these are the partitions as described in Lemma 2.

If (i), (ii) are both satisfied and $t = 1$, then again by (ii), either $G_R = (A, B', A \times B')$ for some $B' \subset B$, or $G_R = (A', B, A' \times B)$ for some $A' \subset A$. In the former case the partitions as described in Lemma 2 are given by $A' = A, A'' = \emptyset, B', B'' = B \backslash B'$; and in the latter case by $A', A'' = A \backslash A', B' = B, B'' = \emptyset$.

When any of the conditions in (i), (ii) are violated, G has a 2×2 odd subgraph which can be found as follows.

If one of the connected components of G_R is not complete bipartite, a 2×2 odd subgraph of G can be found with at most $O(n^2)$ effort as described in the proof of Theorem 5.

If each of the connected components of G_R is a complete bipartite subgraph of G, but their number $t \geq 3$, let $I_v \subset A, J_v \subset B$ be the sets of nodes in the vth connected component K_v of G_R for $v = 1$ to t. Select any node $u \in I_1, p \in J_1, w \in I_2, q \in J_3$. The 2×2 subgraph induced by $\{u, w, p, q\}$ has 1 red and 3 blue edges.

Suppose $t = 2$, and these connected components are the complete bipartite subgraphs $(I_1, J_1, I_1 \times J_1)$ and $(I_2, J_2, I_2 \times J_2)$, where $I_v \subset A, J_v \subset B$ for $v = 1, 2$. If $I_1 \cup I_2 \neq A$, select any $u \in I_1, p \in J_1, q \in J_2, w \in A \backslash (I_1 \cup I_2)$. If $I_1 \cup I_2 = A$, but $J_1 \cup J_2 \neq B$, select any $u \in I_1, p \in J_1, q \in B \backslash (J_1 \cup J_2), w \in I_2$. The 2×2 subgraph of G induced by $\{u, w, p, q\}$ has 1 red and 3 blue edges.

Suppose $t = 1$, and G_R is the complete bipartite subgraph $(I_1, J_1, I_1 \times J_1)$ where $I_1 \subset A, J_1 \subset B$. If $I_1 \neq A$ and $J_1 \neq B$, select any $u \in I_1, p \in J_1, w \in A \backslash I_1, q \in B \backslash J_1$. The 2×2 subgraph induced by $\{u, w, p, q\}$ has 1 red and 3 blue edges.

Clearly the effort needed to check whether conditions (i), (ii) hold; to find the partitions as described in Lemma 2 if these conditions hold; or to find a 2×2 odd subgraph of G when any of these conditions are violated; is at most $O(n^2)$. ∎

Theorem 7. *Let α = the cardinality of a maximum cardinality matching in G_R. The necessary and sufficient conditions for G to have a solution matching for the requirement vector $[n-1, 1]$ are that either $\alpha = n-1$; or that $\alpha = n$ and that G have a 2×2 odd subgraph with 1 blue and 3 red edges.*

Proof. Clearly, $\alpha \geq n-1$ is a necessary condition for the existence of a solution matching in G for the requirement vector $[n-1, 1]$.

If $\alpha = n-1$, let $\bar{M}$ be any maximum cardinality matching in G_R. So, $|\bar{M}| = n-1$. Let A_p, B_q be the exposed nodes in G with respect to the matching $\bar{M}$. If (A_p, B_q) is red, then $\bar{M} \cup \{(A_p, B_q)\}$ is a perfect matching in G_R, contradicting the hypothesis that $\alpha = n-1$. So, (A_p, B_q) must be a blue edge, and $\bar{M} \cup \{(A_p, B_q)\}$ is a solution matching for the requirement vector $[n-1, 1]$.

Now consider the case where $\alpha = n$. Suppose G has no 2×2 odd subgraph containing 1 blue and 3 red edges. Lemma 5 and the fact that G has a red perfect matching imply that there exist partitions $A = A^1 \cup \ldots \cup A^k, B = B^1 \cup \ldots \cup B^k$, $k > 2$ such that $A^t \neq \emptyset, B^t \neq \emptyset$ for all $t = 1$ to k, and all the edges in (A^t, B^t) are red and all the other edges are blue. So, after rearranging nodes so that the rows and columns in the array representation appear as in Figure 7, the main diagonal represents a red perfect matching. These facts imply that any perfect matching in G that contains one edge outside of $\cup_{t=1}^{k}(A^t, B^t)$ must also contain at least one more edge outside this union. Hence in this case there exists no solution matching in G for the requirement vector $[n-1, 1]$.

The only remaining case to consider is when $\alpha = n$ and G has a 2×2 subgraph, $\hat{G}$ say, with 1 blue and 3 red edges. Let $\bar{M}$ be any matching of cardinality $n-1$ in G_R. Let A_p, B_q be the exposed nodes in G with respect to $\bar{M}$. If (A_p, B_q) is blue, then $\bar{M} \cup \{(A_p, B_q)\}$ is a desired solution matching, we are done.

If (A_p, B_q) is red, $\bar{M} \cup \{(A_p, B_q)\}$ is a red perfect matching in G. Let N be the set of all nodes which include the nodes of $\hat{G}$, and all the nodes on matching edges in $\bar{M} \cup \{(A_p, B_q)\}$ incident to nodes of $\hat{G}$. Clearly $|N \cap A| = |N \cap B|$ and $4 \leq |N| \leq 8$. Let $n'' = |N|/2$. Let G'' be the complete bipartite graph of order $n'' \times n''$ which is the subgraph of G induced by N. Then we will show below that G'' has a perfect matching M'' with exactly 1 blue edge and other edges red. Let M' be the set of all matching edges in $\bar{M} \cup \{(A_p, B_q)\}$ that are not incident to any node in N. Then $M'' \cup M'$ is a solution matching in G for the requirement vector $[n-1, 1]$.

Now to show that G'' must have a perfect matching with exactly one blue edge. If $n'' = 2$, then $G'' = g'$, in this case one of the two perfect matchings in G'' has exactly one blue edge.

If $n'' = 3$, rearrange the nodes of G'' so that in the array form of G the main diagonal has the red perfect matching. In this case if the colors of the other cells in the array are not symmetric about the main diagonal, this case can easily be reduced to the case of $n'' = 2$. So, the only form of G'' (equivalent under rearrangement) left to consider is the symmetric case with colors of cells as shown in the following array,

R	R	R
R	R	B
R	B	R

and this array can be verified to have a perfect matching with 1 blue and 3 red edges.

If $n'' = 4$, rearrange the nodes of G'' so that in the array representation the main diagonal contains the red perfect matching. Many cases reduce easily. The only one that is not trivially reduced corresponds to the array form

R		R	R
	R	B	R
R	B	R	
R	R		R

in which the blank cells may be red or blue. In this case also, it can be verified that there is a perfect matching with exactly one blue edge.

Thus in all cases we have verified that G'' has a perfect matching with exactly one blue edge. ∎

7 Algorithm for the General Problem

The statement of our original algorithm was long and tedious to read. We are grateful to a referee who suggested a much simpler way of presenting it. We present this improved version.

Step 1:

If $r = 0$ or n, the problem is a standard bipartite matching problem (only one color) [3]. Also, among the values 1, $n-1$ for r we consider only $r = n-1$ (if $r = 1$, just interchange the red and blue colors). So we assume that $2 \leq r \leq n-1$. We also assume that $n \geq 5$.

If $2 \leq r \leq n-2$ go to Step2. If $r = n-1$ go to Step 3.

Step 2:

Check whether G has a 2×2 odd subgraph using the conditions described in Theorem 6. If G has no 2×2 odd subgraph; let the partitions as described in Lemma 2 be $A = A' \cup A'', B = B' \cup B''$; use the algorithm for the partitioned case discussed in Section 3 to find a solution matching or conclude that none exists; and terminate.

Otherwise let the 2×2 odd subgraph found in G be $\hat{G}$.

Find a matching of cardinality r in G_R, and a matching of cardinality $n-r$ in G_B. If either of these matchings do not exist, there is no solution matching, terminate. Otherwise, beginning with the matchings obtained in G_R, G_B, use Procedures 1, 2, 3 of Section 2 to obtain red and blue matchings $\bar{M}_R, \bar{M}_B$ satisfying $|\bar{M}_R| = r, |\bar{M}_B| = n-r$, such that $\bar{M}_R \cup \bar{M}_B$ contains either 0 or 1 bad nodes. If the former case $\bar{M}_R \cup \bar{M}_B$ is a solution matching, terminate. Otherwise continue.

Let N be the set of nodes which includes the nodes of $\hat{G}$, the exposed node, and all the nodes that are incident with a matching edge in $\bar{M}_R \cup \bar{M}_B$ incident with the bad node or a node of $\hat{G}$. Clearly $|N \cap A| = |N \cap B|$ and $|N| \leq 12$.

Let M' be the set of all matching edges in $\bar{M}_R \cup \bar{M}_B$ that are not incident to any node in N. Let $r' = r - |M' \cap M_R|, s' = n - r - |M' \cap M_B|$.

If both r', s' are ≥ 2, define $\bar{M} = M', N'' = N$.

If $r' = 1$ [$s' = 1$] select any of the matching edges in $M' \cap M_R$ [$M' \cap M_B$], e say, and let N'' = the union of N and the set of two nodes on the edge e, and $\bar{M} = M' \backslash \{e\}$.

So, $|N''| \leq 14$ and even. Let $n'' = |N''|/2$. Let G'' be the subgraph of G induced by N''. G'' is a complete bipartite graph of order $n'' \leq 7$.

Let M''_R, M''_B be the set of matching edges in M_R, M_B in G'', then $r'' = |M''_R| \geq 2, s'' = |M''_B| \geq 2$.

By Theorem 3, G'' has a perfect matching satisfying the requirement vector $[r'', s'']$. Find it by enumeration, let it be $\bar{M}''$. Then $\bar{M} \cup \bar{M}''$ is a solution matching in the original graph G, terminate.

Step 3:

Find a maximum cardinality matching, $\tilde{M}$ say, in G_R. Let $\alpha = |\tilde{M}|$.

If $\alpha \leq n-2$, there is no solution matching in G, terminate.

If $\alpha = n-1$, let p, q be the exposed nodes in G with respect to the matching $\tilde{M}$. Then from Theorem 7, $\tilde{M} \cup \{(p,q)\}$ is a solution matching in G, terminate.

If $\alpha = n$, find a 2×2 odd subgraph of G with 1 blue and 3 red edges using the procedure described in Theorem 5. If such an odd subgraph does not exist, there exists no solution matching in G, terminate.

Otherwise, let $\hat{G}$ be the 2×2 odd subgraph found. Let N be the set of nodes which includes the nodes of $\hat{G}$ and all the nodes on the matching edges in $\tilde{M}$ incident to nodes in $\hat{G}$. Let G'' be the complete bipartite subgraph of G induced by N. By, Theorem 7 there exists a perfect matching in G'' with exactly one blue edge, find it, M'' say, by enumeration. Let $\tilde{M}'$ be the set of matching edges in $\tilde{M}$ that are not incident to any node in N. Then $\tilde{M}' \cup M''$ is a solution matching in the original graph G, terminate.

8 Computational complexity analysis

The major work in the algorithm is that of finding matchings of cardinalities $r, n-r$ in G_R, G_B respectively, which has complexity $O(n^{2.5})$. The other work takes less time than this. Thus the overall complexity of the algorithm to find a solution matching or establish that there is none is $O(n^{2.5})$.

9 Acknowledgements

We thank the referees for many suggestions. In particular, one referee pointed out an omission in our original proof of Theorem 3 and the simpler proof of Lemma 2, and another suggested the simpler way of presenting the algorithm, for which we are very grateful.

10 References

[1] R. Chandrasekaran, S.N. Kabadi, and K. G. Murty, "Some NP-Complete Problems in Linear Programming", *O. R. Letters*, 1(July 1982)101-104.

[2] A. Gupta and J. Sharma, "Tree Search Method for Optimal Core Managament of Pressurized Water Reactors", *Computers and operations Research*, 8(1981)263-266.

[3] J. E. Hopcroft, and R. M. Karp, "A $n^{5/2}$ algorithm for Maximum Matchings in Bipartite Graphs", *SIAM Journal on Computing*, 2(1973)225-231.

[4] A. V. Karzanov, "Maximum matching of given weight in complete and complete bipartite graphs", *Kibernetika* 1 (Jan. - Feb. 1987)7-11 [English translation in: CYBNAW 23 (1) (1987) 8-13].

[5] K. G. Murty, *Network Programming* (Prentice Hall, Englewood Cliffs, NJ, 1992).

[6] C. Papadimitriou, "Polytopes and Complexity", pp 295-305 in W. R. Pulleyblank (ed.), *Progress in Combinatorial Optimization*, (Academic Press Canada, Don Mills, 1984).

[7] T. Yi, "Bipartite Matchings with Specified Values for a $0{-}1$ Linear Function", Doctoral Dissertation, University of Michigan (Ann Arbor, MI, 1994).