

PERFECTLY MATCHABLE SUBGRAPH PROBLEM ON A BIPARTITE GRAPH

FIRDOVSI SHARIFOV¹

Abstract. We consider the maximum weight perfectly matchable subgraph problem on a bipartite graph $G = (UV, E)$ with respect to given nonnegative weights of its edges. We show that G has a perfect matching if and only if some vector indexed by the nodes in UV is a base of an extended polymatroid associated with a submodular function defined on the subsets of UV . The dual problem of the separation problem for the extended polymatroid is transformed to the special maximum flow problem on G . In this paper, we give a linear programming formulation for the maximum weight perfectly matchable subgraph problem and propose an $O(n^3)$ algorithm to solve it.

Keywords. Bipartite graph, extended polymatroid, perfect matching, perfectly matchable subgraph.

Mathematics Subject Classification. 05C70, 05C85.

1. INTRODUCTION

Cornaz and Mahjoub investigated the maximum induced bipartite subgraph problem on a given graph with nonnegative edge weights [6]. They gave an integer linear programming formulation for this problem and proved that it is *NP*-hard even for the unit case of edge weights. Here we consider a special case of this problem in a given bipartite graph with nonnegative weights on the edges, which is to find a maximum weight induced subgraph having a perfect matching. We give a polynomial time algorithm for this problem. We use standard graph terminology

Received June 5, 2009. Accepted October 20, 2009.

¹ Glushkov Institute of Cybernetics, Glushkova pr. 40, MSP, 252650 Kyiv, Ukraine;
F-sharifov@yandex.ru

and begin by introducing some necessary additional notations. In any graph $G = (V, E)$, the set of edges with one end in S and other end in T is denoted by $[S, T]$ for $S, T \subseteq V$. We use $\gamma(S)$ and $\delta(S)$ to denote the sets $[S, S]$ and $[S, V \setminus S]$, respectively. For a set $L \subseteq V$ or $L \subseteq E$ and a vector $y \in R^V$ or $y \in R^E$, we use $y(L)$ for $\sum(y_v, v \in L)$. A subgraph with the edge set $\delta(w)$ is called a *star* and the node $w \in V$ is called *center* of the star. We define $\gamma(\emptyset) = \emptyset$ and $\delta(\emptyset) = \emptyset$. A node set S induces the subgraph $G(S, \gamma(S))$ of G , and if $G(S, \gamma(S))$ has a perfect matching, then the subset S is a matchable set and $G(S, \gamma(S))$ is a perfectly matchable subgraph of G . We denote by $PMS(G)$ the convex hull of incidence vectors of matchable sets of G .

Balas and Pulleyblank gave the linear description of $PMS(G)$ with respect to any graph G [3,4]. Cunningham and Green-Krotki in [7] showed that $PMS(G) = \{Bz : z \in MP(G)\}$, where $MP(G)$ is the convex hull of incidence vectors of matchings of the graph G and B is the $|V| \times |E|$ incidence matrix of G . It means that for any vector $c' = (c'_v : v \in V)$ of node weights

$$\max \{c'x : x \in PMS(G)\} = \max \{(c'B)z : z \in MP(G)\}.$$

So, with respect to any node weights c'_v , a maximum matchable set of G can be found by solving the well known maximum weight matching problem on G with the weights $c'_{uv} = c'_u + c'_v$ on the edges $(u, v) \in E$. Cunningham and Green-Krotki also proposed an polynomial time algorithm for the separation problem for $PMS(G)$ and a given $|V|$ -vector x . This algorithm maintains feasibility z ($z \in MP(G)$) for $x = Bz$. Moreover the vector z satisfies certain additional conditions that are required in the separation problem. It is not necessary that a solution of the right-hand side of the above equality defines a perfect matching. Hence, it makes us unable to find a matchable set inducing a maximum weight matchable subgraph required in the following closely related problem by setting $c' = 1$ in the right-hand side of the above equality.

In a given bipartite graph $G = (UV, E)$ ($UV = U \cup V$) with nonnegative edge weights c_e for $e \in E$, it is required to find a matchable set S such that $c(\gamma(S))$ is maximum and the subgraph $G(S, \gamma(S))$ is perfectly matchable. We call this the maximum weight perfectly matchable subgraph problem (PMSM). We propose an $O(n^3)$ algorithm for this problem. Our algorithm first finds $UV_0 \subseteq UV$ such that any maximum cardinality matching covers all nodes of UV_0 . It allows us to modify the edge weights so that we can use the above linear relationship to find an optimal matchable set to the PMSM. That is, the nodes covered by maximum weight bipartite matching constitute an optimal matchable set. Notice that the set UV_0 cannot be found by solving the right-hand side of the above equality when $c' = 1$ since there is a maximum cardinality matching covering a node $w \notin UV_0$.

We define the functions $f^*(S) = |\gamma(S) \cup \delta(S)|$ and $f(S) = |\gamma(S)|$ for all $S \subseteq UV$. It is known that $f^*(S)$ is a submodular and $f(S)$ is a supermodular function [8]. In Section 2, we show that the bipartite graph $G = (UV, E)$ has a perfect matching if and only if the vector $\sigma = (-b_{u_1}, \dots, -b_{u_n}, b_{v_1}, \dots, b_{v_n})$ is a base of the extended polymatroid $P(f^* - f)$ associated with the submodular function $f^*(S) - f(S)$ where

$b_{u_i} = d_{u_i} - 1$, $b_{v_i} = d_{v_i} - 1$ and d_{u_i} , d_{v_j} are degrees of nodes $u_i \in U$ and $v_i \in V$. Moreover, this can be regarded as a generalization of the following result due to Balinski [5]. A spanning tree \mathcal{T} of a bipartite graph G has a perfect matching if $t = (t_{u_1}, \dots, t_{u_n})$ is any vector of the form $(2, \dots, 2, 1, 2, \dots, 2)$, where t_{u_i} is the degree of a node $u_i \in U$ with respect to a spanning tree \mathcal{T} . The vector t is called a signature of the tree \mathcal{T} . We also call the vector σ a signature of the bipartite graph G . The problem of the testing whether σ is a base of $P(f^* - f)$ is equivalent to the separation problem; whether σ belong to $P(f^* - f)$ or not (see the proof of Thm. 1). A dual problem of the optimization problem is transformed to the maximum flow problem on the network with unit capacity of its edges. In Section 3, we describe the Breadth-First Search simple augmentation path algorithm for this maximum flow problem.

In Section 4, we consider the PMSP for which Sections 2 and 3 contain background materials. We formulate the PMSP as a linear programming problem and show that the set UV_0 can be found by solving the maximum flow problem. It allows us to modify the edge weights so that we can apply any maximum weight matching algorithm to find maximum weight matchable subgraph in a bipartite graph $G = (UV, E)$ with nonnegative weight on the edges.

2. PERFECT MATCHING AND EXTENDED POLYMATROID

In this section, we prove some new results on the existence of a perfect matching in a bipartite graph. The perfect bipartite matching problem is a maximum cardinality matching problem on a bipartite graph $G = (UV, E)$ for which $|U| = |V|$. The problem of finding a maximum cardinality matching on any bipartite graph $G' = (UV', E')$ ($UV' = U' \cup V'$) can be reduced to the same problem on the graph $G = (UV, E)$ for which $|U| = |V|$. Suppose that $|U'| < |V'|$. Then we add $|V'| - |U'|$ new nodes to U' and connect each new node with each node in V' by new edges. For the graph $G = (UV, E)$ with $U = U'$ and $V = V'$ we have $|U| = |V|$. Let M_G be a maximum cardinality matching in G and M be a submatching of M_G such that the endpoints of its edges are new nodes.

Proposition 2.1. *If M_G is a maximum cardinality matching in G , then $M_G \setminus M$ is a maximum cardinality matching in G' .*

Proof. Let M' be a maximum cardinality matching in G' and $|M'| > |M_G \setminus M|$. Since each new node is connected with each node $v \in V$, we can find the matching M_0 in G , so that M_0 contains M' and some submatching covering all new nodes. Then $|M'| > |M_G \setminus M|$ implies $|M_0| > |M_G|$. This contradicts the assumption that M_G is a maximum cardinality matching in G . \square

Thus, in Sections 2 and 3, without loss of generality, we assume that $|U| = |V| = n$ for the bipartite graph $G = (UV, E)$. Consider the polytope

$$P(f^* - f) = \{x \in R^{UV} : x(S) \leq f^*(S) - f(S), S \subseteq UV\}$$

which is called an extended polymatroid in [8] associated by the submodular function $f^* - f$. Any vector $x \in R^{UV}$ is called a base of $P(f^* - f)$ if $x(UV) = f^*(UV) - f(UV)$ and $x(S) \leq f^*(S) - f(S)$ for all $S \subseteq UV$. Since $f^*(UV) = f(UV)$ by the definitions of f^* and f , then $x(UV) = 0$ for any base x of $P(f^* - f)$. For the linear order $\{w_1, \dots, w_{2n}\}$ of nodes w_i in UV , the corresponding base $x^0 = (x_{w_1}^0, \dots, x_{w_{2n}}^0)$ can be found by the greedy algorithm [8]:

$$x_{w_i}^0 = f^*(L_i) - f(L_i) - f^*(L_{i-1}) + f(L_{i-1}), \quad i = 1, \dots, 2n \quad (1)$$

where $L_0 = \emptyset$ and $L_i = \{w_1, \dots, w_i\}$ for all $i = 1, \dots, 2n$. For the linear orders $\{u_1, \dots, u_n, v_1, \dots, v_n\}$ and $\{v_1, \dots, v_n, u_1, \dots, u_n\}$ of the nodes $u_i \in U$ and $v_i \in V$, formula (1) defines the vectors

$$x^d = (d_{u_1}, \dots, d_{u_n}, -d_{v_1}, \dots, -d_{v_n})$$

and $-x^d$ which are bases of $P(f^* - f)$.

Theorem 2.2. *A bipartite graph G has a perfect matching if and only if its signature σ is a base of the extended polymatroid $P(f^* - f)$.*

Proof. By Proposition 2.1 we have $|U| = |V|$, from which it follows that $\sigma(UV) = 0$. Hence, to prove the theorem, we need to show that if $\sigma \in P(f^* - f)$ then, G has a perfect matching and conversely.

Let us show that if $\sigma \in P(f^* - f)$, then G has a perfect matching. From the definition of the signature σ , it follows that

$$d(N_A) - d(A) - |N_A| - |A| = (b(N_A) - b(A)) = \sigma(A \cup N_A)$$

for each $A \subseteq U$, where N_A denotes the set of nodes adjacent to some node in A . By the definitions of the functions f^* and f , we have

$$d(N_A) - d(A) = f^*(A \cup N_A) - f(A \cup N_A).$$

Since $\sigma \in P(f^* - f)$, then

$$f^*(A \cup N_A) - f(A \cup N_A) - \sigma(A \cup N_A) \geq 0,$$

from which it follows that

$$\begin{aligned} |N_A| - |A| &= d(N_A) - d(A) - (b(N_A) - b(A)) \\ &= f^*(A \cup N_A) - f(A \cup N_A) - \sigma(A \cup N_A) \geq 0. \end{aligned}$$

Thus, $|N_A| \geq |A|$ for any $A \subseteq U$. From $|U| = |V|$ it follows that the graph G has a perfect matching by the well known Frobenius Marriage theorem.

Now, let M be a perfect matching in G . We must show that $\sigma \in P(f^* - f)$. Consider the bipartite graph $G_1 = (UV, E \setminus M)$, that is G_1 is the subgraph defined by the edge set $E \setminus M$. We can define the submodular function $f_1^*(A)$ and the

supermodular function $f_1(A)$ for the graph G_1 as we have defined $f^*(A)$ and $f(A)$ for the graph G . It is clear that $f_1^*(A) - f_1(A) \leq f^*(A) - f(A)$ for any $A \subseteq UV$. Moreover, b_w is the degree of a node w of the graph G_1 . Hence, the vector $-x_1^b = (-b_{u_1}, \dots, -b_{u_n}, b_{v_1}, \dots, b_{v_n})$ similar to base $-x^d$ of $P(f^* - f)$, is a base of the extended polymatroid $P(f_1^* - f_1)$. From $\sigma = -x_1^b$, it implies that $\sigma(A) \leq f_1^*(A) - f_1(A) \leq f^*(A) - f(A)$ for any $A \subseteq UV$ and this completes the proof of the theorem. \square

Corollary 2.3. σ is a base of the extended polymatroid $P(f^* - f)$ if and only if $-\sigma$ is a base of it.

Theorem 2.2 states that the graph G has a perfect matching if and only if the value

$$\min \{f^*(A) - f(A) - \sigma(A) : A \subseteq UV\} \quad (2)$$

is not negative. The minimum value of this problem can be found in $O(|UV||E|)$ time by the algorithm in [9]. Here we propose a simpler algorithm than the one in [9].

From the definitions of the functions f^* and f , we have $f^*(S) + f(S) = d(S)$ for any $S \subseteq UV$, where $d = (d_w : w \in UV)$ and d_w is the degree of a node $w \in UV$. Hence, problem (2) is transformed to find

$$\min \{2f^*(A) - [d(A) + \sigma(A)] : A \subseteq UV\}.$$

Since $d_w + \sigma_v > 0$ for all $w \in UV$ and $f^*(A)$ is a monotone submodular function, then the dual problem of the latter one can be written as follows [8]:

$$\max \{x(UV) : x(A) \leq 2f^*(A), A \subseteq UV, 0 \leq x \leq d + \sigma\}$$

which is transformed to

$$\max \{y(UV) : y \in P(f^* - f), -d \leq y \leq \sigma\}, \quad (3)$$

for $y = x - d$. In (3), we can replace the inequalities $-d_u \leq y_u \leq b_u$ by $0 \leq y_u \leq b_u$ for the nodes $u \in U$ since such replacing does not change an optimal solution to (3). Now let

$$y_u = z(\delta(u)), \quad y_v = -z(\delta(v)) \quad \text{for all } u \in U, \quad v \in V.$$

If we restrict the new variables z_e to

$$0 \leq z_e \leq 1, \quad e \in E,$$

then the conditions $0 \leq y_u \leq b_u$ and $-d_v \leq y_v \leq -b_v$ can be written as follows:

$$z(\delta(u)) \leq b_u, \quad u \in U, \quad z(\delta(v)) \geq b_v, \quad v \in V. \quad (4)$$

By Corollary 2.3, we can obtain problem (2) with respect to $-\sigma$. For $-\sigma$ problem (3) contains the conditions $-d_u \leq y_u \leq -b_u$ and $0 \leq y_v \leq b_v$. In the same

way, it can be shown that these conditions are equivalent to

$$z(\delta(u)) \geq b_u, \quad u \in U, \quad z(\delta(v)) \leq b_v, \quad v \in V. \quad (5)$$

From (4) and (5), it implies that the vector $z = (z_e : e \in E)$ is a solution to the following linear system of equations on the bipartite graph G :

$$z(\delta(w)) = b_w, \quad w \in UV, \quad (6)$$

$$0 \leq z_e \leq 1, \quad e \in E. \quad (7)$$

Note that for some $w \in UV$ if $b_w = 0$ then $z_e = 0$, where e is an edge with one of its endpoints w , that is $\delta(w) = \{e\}$.

Theorem 2.4. *The signature σ is a base of $P(f^* - f)$ if and only if there exists a feasible solution to the linear system (6)–(7) on a bipartite graph $G = (UV, E)$.*

Proof. Suppose that there exists a feasible solution to (6)–(7). Then, there exists an integer 0 and 1 basis feasible solution $z = (z_e : e \in E)$ to (6)–(7). Let

$$E_1 = \{e : z_e = 1, e \in E\}.$$

From $b_w = d_w - 1$ for any $w \in UV$, it follows that any node has a unit degree in the subgraph defined by the subset $E \setminus E_1$. Since $|U| = |V|$, the edge set $E \setminus E_1$ is a perfect matching in G . Thus, $\sigma \in P(f^* - f)$ by Theorem 2.2.

Conversely, let $\sigma \in P(f^* - f)$. By Theorem 2.2, the graph G has a perfect matching M . Then, b_w is the degree of a node $w \in UV$ in the subgraph obtained by deleting all edges of M from G . Since $b(U) = b(V)$ for the right-hand side vector $b = (b_{u_1}, \dots, b_{u_n}, b_{v_1}, \dots, b_{v_n})$, we can define a solution of (6)–(7) by setting $z_e = 1$ for $e \in E \setminus M$ and $z_e = 0$ for $e \in M$. \square

Corollary 2.5. *A bipartite graph G has a perfect matching if and only if there exists a feasible solution to the linear system (6)–(7).*

By Corollary 2.5, problem 2 is equivalent to finding a solution of the linear system (6)–(7) on the graph G . It is clear that if there exists a solution of (6)–(7), it is an optimal solution to the following problem on the graph G :

$$m^* = \max \sum_{e \in E} z_e, \quad (8)$$

$$z(\delta(w)) \leq b_w, \quad w \in UV, \quad (9)$$

$$0 \leq z_e \leq 1, \quad e \in E. \quad (10)$$

It is well known that any basis solution of (6)–(7) (or (9)–(10)) is 0 and 1 integer valued and it is the incidence vector of some set of edges. Using this set of edges, a maximum cardinality matching can be easily found in the graph G . The following lemma shows how to do it.

Lemma 2.6. *Let $z = (z_e : e \in E)$ be an optimal basis solution to the problem (8)–(10) and let G_0 be a subgraph obtained by deleting all edges e from G such that $z_e = 1$, i.e. G_0 contains edges e for which $z_e = 0$. Then, G_0 consists of a matching M_0 and some stars.*

Proof. Since z is an optimal basis solution to the problem (8)–(10), the graph G has no edges $e = (u, v)$ such that $z(\delta(u)) < b_u$, $z(\delta(v)) < b_v$ and $z_e = 0$. Thus, if $z(\delta(u)) < b_u$ and $z(\delta(v)) < b_v$ then either the edge e is not in G or $z_e = 1$. Hence, G_0 has no edge $e = (u, v)$ such that $z(\delta(u)) < b_u$, $z(\delta(v)) < b_v$. Now, for any edge (u, v) of G_0 , if $z(\delta(u)) = b_u$, then either $z(\delta(v)) = b_v$ (in this case, the nodes u and v are covered by the matching M_0) or $z(\delta(v)) < b_v$ (in this case, there is a star with center v in G_0). \square

Let $z = (z_e : e \in E)$ be an optimal basis solution to (8)–(10). If the condition (9) holds as equality for all $w \in UV$, then G_0 is a perfect matching in G . Otherwise, (the linear system (6)–(7) has no solution), by Lemma 2.6, the graph G_0 consists of stars and a matching M_0 . Let M be a matching that contains M_0 and one edge of each star in G_0 . Since z is an optimal basis solution, it is easy to see that M is a maximum cardinality matching in G . Hence, if one has the graph G_0 , a maximum matching can be found in $O(n)$ time. In the next section, we present an $O(mn)$ ($m = |E|$) algorithm to solve (8)–(10) or equivalently to get the graph G_0 .

3. AN ALGORITHM FOR THE PROBLEM (8)–(10)

To solve the problem (8)–(10), first the following procedure Marking marks edges of some subset of E so that the incidence vector of the marked edges is a feasible solution to (8)–(10). Then the Breadth-First Search procedure checks the optimality of this solution and makes it an optimal if it is not an optimal solution.

Procedure Marking;

- 1. Scan a node $u \in U$ and if it is possible, mark a maximum number (but not greater than b_u) of edges (u, v) with $b_v > 0$.**
- 2. Repeat step 1 for all nodes $u \in U$.**

Let $MarkU = \{u : z(\delta(u)) < b_u, u \in U\}$, $MarkV = \{v : z(\delta(v)) < b_v, v \in V\}$ and let $z = (z_e : e \in E)$ be the incidence vector of the marked edges. It is clear that the vector z is a feasible solution to (8)–(10) and if $MarkU = \emptyset$ or $MarkV = \emptyset$ then z is an optimal solution. If $MarkU \neq \emptyset$ or $MarkV \neq \emptyset$, we can not be convinced of the optimality of z . In this case, we try to make z an optimal solution. For that, we iterate the Breadth-First Search procedure to find an augmenting path between any pairs of nodes $u \in MarkU$ and $v \in MarkV$ on the graph with arcs: all marked edges (u, v) are replaced by arc (v, u) and unmarked edges (u, v) are replaced by arc (u, v) , where $u \in U$ and $v \in V$.

Breadth-First Search Procedure;

begin(installation parameters)

```

    level = 0;
    label(u) = 0 for some u ∈ MarkU;
    label(w) = n for all w ∈ UV \ {u};
    n_w = z(δ(w)) for all w ∈ UV;
end(begin)
begin(finding path)
1. if an arc (u, v) is unmarked and label(u) = level    label(v) := level;
2. If v ∈ MarkV (the path from u to v is found)
    then do
        set the marked arcs of the path as unmarked
        and mark the unmarked arcs of the path;
        n_u := n_u - 1; n_v := n_v - 1;
        if n_u = 0, n_v = 0 then delete u from MarkU, v from MarkV;
        stop;
    end(do)
else if an arc (v, u) is marked and label(v) = level
    then do
        label(u) := level + 1;
        goto 1;
    end(do)
end(begin)

```

If the Breadth-First Search procedure finds a path between any pair of nodes $u \in \text{Mark}U$ and $v \in \text{Mark}V$, then it is an augmenting path and the incidence (characteristic) vector z of the set of marked edges is a feasible solution of (8)–(10). If a value of $label(w)$ for some node $w \in UV$ is not increasing at some iteration, this means that the incidence vector z of the current marked edges is an optimal solution. We abbreviate this algorithm as AM&P (Algorithm Marking and Perfect).

Lemma 3.1. *The Procedure Marking marks at least $m - 2n + 2$ edges.*

Proof. Similarly to the proof of Lemma 2.6, it is easy to see that the Procedure Marking returns the set of unmarked edges which contains a matching and stars. This set forms a forest with $2n$ nodes. A number of edges in a forest is maximum when it consists of two trees. Hence the number of unmarked edges is at most $2(n - 1)$. Therefore, the number of marked edges is at least $m - 2n + 2$. \square

By Lemma 3.1 $m - 2n + 2 \leq m^* \leq b(U) = m - n$. Hence, $m - n - m^* \leq n - 2$ from which it follows that the Breadth-First Search procedure finds at most $n - 2$ augmenting paths. Thus, the AM&P algorithm runs in $O(mn)$ time. In bipartite graphs for which $m = O(n)$, the finding of a maximum cardinality matching is more arduous than in other graphs. The algorithm AM&P runs in $O(n^2)$ time for $m = O(n)$ and it is a suboptimal algorithm. Up to now, the best bipartite maximum cardinality matching algorithm [2] runs in $O(n^{1.5} \sqrt{m/\log|V|})$ time.

4. PERFECTLY MATCHABLE SUBGRAPH PROBLEM ON A BIPARTITE GRAPH

Let $G = (UV, E)$ be any bipartite graph where we do not require that $|U| = |V|$. It is known that $S \subseteq UV$ induces a perfectly matchable subgraph $G(S, \gamma(S))$ if it has a perfect matching. Let $x = (x^1, x^2)$ ($x^1 = (x_u^1, u \in U)$, $x^2 = (x_v^2, v \in V)$) and $z = (z_e : e \in E)$ be vectors of node and edge variables, respectively. Let \mathcal{P} denote the set of all perfectly matchable subgraphs of G . We assume that $G(\emptyset, \gamma(\emptyset))$ is in \mathcal{P} and let $G(S, \gamma(S))$ be in \mathcal{P} . Then, we show that the 0 or 1 incidence vectors x and z of the sets S , $\gamma(S)$ satisfy the following linear system:

$$z(\delta(u)) - x_u^1 = y(\delta(u)), \quad u \in U, \quad (11)$$

$$z(\delta(v)) - x_v^2 = y(\delta(v)), \quad v \in V, \quad (12)$$

$$z_e \leq x_u^1, \quad u \in U, \quad e = (u, v) \in E, \quad (13)$$

$$z_e \leq x_v^2, \quad v \in V, \quad e = (u, v) \in E, \quad (14)$$

$$x^1(U) - x^2(V) = 0, \quad (15)$$

$$0 \leq x_v \leq 1, \quad v \in UV, \quad (16)$$

$$0 \leq z_e \leq 1, \quad 0 \leq y_e \leq 1, \quad e \in E. \quad (17)$$

Since z is the incidence vector of $\gamma(S)$, then $z(\delta(u))$ and $z(\delta(v))$ are the degrees of nodes $u \in S \cap U$ and $v \in S \cap V$ in the subgraph $G(S, \gamma(S))$. Hence, it follows that $z(\delta(S \cap U)) = z(\delta(S \cap V))$. Since $G(S, \gamma(S))$ has a perfect matching, then $x^1(U) = x^2(V)$, from which it follows that

$$b(S \cap U) = z(\delta(S \cap U)) - x^1(U) = z(\delta(S \cap V)) - x^2(V) = b(S \cap V),$$

for $b_u = z(\delta(u)) - x_u^1$ and $b_v = z(\delta(v)) - x_v^2$. Hence, by Theorem 2.4 the linear system (11)–(17) has a feasible solution.

Now, we want to show that any basis solution of (11)–(17) is 0 or 1 integer valued. Let B denote the $(|U| + |V|) \times |E|$ incidence matrix of the bipartite graph G and let B_1 and B_2 denote the submatrices of B consisting of the rows corresponding to the nodes in U and the nodes in V , respectively. Then, the constraints (11)–(15) can be rewritten as follows:

$$Bz - I_{12}x - By = 0,$$

$$Iz - B_1^t x^1 \leq 0,$$

$$Iz - B_2^t x^2 \leq 0,$$

$$\mathbf{1}_1 x^1 - \mathbf{1}_2 x^2 = 0,$$

where I and I_{12} are the $|E| \times |E|$ and $(|U| + |V|) \times |E|$ identity matrices, B_1^t and B_2^t are transpose of matrices B_1 and B_2 . Moreover, $\mathbf{1}_1 = (1, \dots, 1)$, $\mathbf{1}_2 = (1, \dots, 1)$ are the $1 \times |U|$ and $1 \times |V|$ matrices (vectors), respectively. Let \mathcal{A} be a quadratic submatrix consisting of some rows and columns that belong to some of the above matrices. Then, the following cases are possible:

1. In \mathcal{A} , there is no row having some entries of the matrices $\mathbf{1}_1$ and $-\mathbf{1}_2$. Since the incidence matrix of any bipartite graph is unimodular, it follows that $\det(\mathcal{A}) = 0$ or ∓ 1 .
2. There is a row l of \mathcal{A} having some entries of the matrices $\mathbf{1}_1$ and $-\mathbf{1}_2$. Then, by adjusting columns of \mathcal{A} using addition and subtraction, we can transform \mathcal{A} to the submatrix \mathcal{B} with the row l consisting of unit nonzero entry. Then $\det(\mathcal{A}) = \det(\mathcal{B})$ by the well-known properties of determinants. By the algebraic complement decomposition of $\det(\mathcal{B})$ with respect to the nonzero entry of the row l , we have $\det(\mathcal{B}) = (-1)^k \det(\mathcal{B}_0)$. Since the matrix \mathcal{B}_0 has no entries of the matrices $\mathbf{1}_1$ and $\mathbf{1}_2$, then $\det(\mathcal{B}_0) = 0$ or ∓ 1 .

Moreover, it is easy to see that (11)–(17) has a feasible solution. Since constraints (16)–(17) are trivial, this implies that there exists 0 or 1 integer valued basis solution of the linear system (11)–(17). Let x , y and z be some basis solution to (11)–(17). Then x is the 0 or 1 incident vector of some $S \subseteq UV$. The conditions (12) and (13) state that z is also the 0 or 1 incident vector of some $E(S) \subseteq \gamma(S)$. Hence, $z(\delta(u))$ and $z(\delta(v))$ are degrees of nodes $u \in U \cap S$ and $v \in V \cap S$ in $G(S, E(S))$, respectively. Then, from condition (12) we have

$$z(\delta(S \cap U)) - x^1(U) = z(\delta(S \cap V)) - x^2(V).$$

By Theorem 2.4, the subgraph $G(S, E(S))$ has a perfect matching. Since $E(S) \subseteq \gamma(S)$, the subgraph $G(S, \gamma(S))$ induced by matchable set S also has a perfect matching.

Now let us consider the perfectly matchable subgraph problem on any bipartite graph. We have abbreviated it as PMSP: given a bipartite graph $G = (UV, E)$ with a weight $c_e \geq 0$ for all $e \in E$ and it is required to find $S \subseteq UV$ such that the subgraph $G(S, \gamma(S))$ has a perfect matching and $c(\gamma(S))$ is a maximum. From the above results, it follows that PMSP can be formulated as the following linear programming problem:

$$\text{maximize } \sum_{e \in E} c_e z_e,$$

subject to (11)–(17).

Lemma 4.1. *Let $G_* = (S_*, \gamma(S_*))$ be a subgraph required in the PMSP and let M_* be a perfect matching in G_* . If $c_e \geq 0$ for all $e \in E$, then $|M| = |M_*|$, where M is a maximum cardinality matching in a bipartite graph G .*

Proof. If $|M| > |M_*|$, then there exist nodes $u \in U$ and $v \in V$ such that they are covered by M and they are not covered by M_* . Hence, by applying the Breadth-First Search procedure, we can find an augmenting path between the nodes u and v . The edges of this path are alternatively in and not in M . We add all edges of this path to $\gamma(S_*)$. Then, a maximum cardinality matching in the graph defined by $\gamma(S_*)$ contains more edges than M_* . Let us denote the maximum cardinality matching again by M_* . If $|M_*| < |M|$, by following the above procedure, we have

$|M| = |M_*|$ and M is a subset of $\gamma(S_*)$. From $c_e \geq 0$ for all $e \in E$ it follows that the value of $c(\gamma(S_*))$ can be just increased. \square

The preceding lemma states that there exists a maximum cardinality matching M in G such that $M = M_*$. It is easy to show by an example that we can not replace M by a maximum weight matching in $M = M_*$. However we can show that the maximum cardinality matching M can be found by applying first any maximum flow algorithm and then any bipartite maximum weight matching algorithm. For this, we create a directed version of the underlying graph G by designating all arc as pointing from the nodes in U to the nodes in V . Then we introduce a source node s and a sink node r , with an arc connecting s to each node in U and an arc connecting each node in V to r . We set capacity of arcs (s, u) and (v, r) to b_u and b_v , respectively. For the arc in E , we assign a unit capacity. This network is denoted by G_{sr} which is called a network with underlying graph G . We have found the maximum flow from s to r in the network G_{sr} . Using the residual capacity on the edges of the residual network corresponding to the maximum flow (see [1]) we can find the minimum $s - r$ cut; if there exists a directed path from source s to a node w , we add the node w to the set A . Then $[A, \bar{A}]$ is a minimum cut and $s \in A$, $r \in \bar{A}$, where $\bar{A} = (UV \cup \{s, r\}) \setminus A$. It is not difficult to see that the cut $[A, \bar{A}]$ can be found by using the graph G_0 (see Lem. 2.6) as follows. Let $\delta_0(w)$ denotes a star with center w in the graph G_0 . If G_0 has no a star, then either $A = \{s\}$ or $\bar{A} = \{r\}$. If there are stars $\delta_0(w)$ in G_0 , then the set A contains leaves of any star and nodes covered by matching M_0 (see Lem. 2.6). For any bipartite graph G following cases are possible.

Case 1. The graph G_0 contains a matching M_0 and stars $\delta_0(w_1), \dots, \delta_0(w_p)$ such that the centers w_1, \dots, w_p are either in V or in U . In this case the cut $[A, \bar{A}]$ is the set either of all arcs (s, u) , $u \in U$ or of all arcs (v, r) , $v \in V$, respectively.

Case 2. The graph G_0 contains a matching M_0 and stars $\delta_0(w_k)$ such that $w_k \in U$ or $w_k \in V$ for $k = 1, \dots, p$.

Lemma 4.2. *If (s, u) ((v, r)) is not an arc of the minimum cut $[A, \bar{A}]$ then in the graph G any maximum cardinality matching covers the node $u \in U$ ($v \in V$).*

Proof. For the case 1 the proof of the lemma is trivial. Thus, consider the case 2. If (s, u) is not an edge of the cut $[A, \bar{A}]$ then either the node $u = w_k$ or $u \neq w_k$ for $k = 1, \dots, p$, where w_k is a center of stars $\delta_0(w_k)$. Consider the case when $u = w_k$ that is u is a center of some star $\delta_0(w_k)$ and $w_k \in U$. Then an amount of the flow on the arc (s, w_k) is less than b_{w_k} and any maximum cardinality matching in G covers the node w_k . Moreover, if v is a leaf of the star $\delta_0(w_k)$ then $[A, \bar{A}]$ contains the arc (v, r) and there exists a maximum cardinality matching covering the node v . Now consider the case when (s, u) is not an arc of $[A, \bar{A}]$ and $u \neq w_k$ for all $k = 1, \dots, p$. Then the node u is covered by the matching M_0 in the graph G_0 (see the proof of Lem. 2.6). Hence the set N_u contains some nodes $\{v_0, \dots, v_h\}$ such that $(u, v_0), (u_1, v_1), \dots, (u_h, v_h)$ are the edges of M_0 . Then $|\{u, u_1, \dots, u_h\}| \geq |N_{u, u_1 \dots u_h}|$. Therefore any maximum cardinality matching in G covers all nodes in $N_{u, u_1 \dots u_h}$,

in particular, the node u . By the same way it proves that if (v, r) is not an arc of the cut $[A, \bar{A}]$ the node v is covered by maximum matching covers in G . \square

From Lemma 4.2, we have the important fact that if (s, u) $((r, v))$ is an arc of $[A, \bar{A}]$, any maximum cardinality matching covers all nodes N_u (N_v). This implies that if (s, u) $((v, r))$ is an arc of $[A, \bar{A}]$ and $u \in S_*$ ($v \in S_*$) then $\delta(u) \subseteq \gamma(S_*)$ ($\delta(v) \subseteq \gamma(S_*)$) since $c_e \geq 0$ for all $e \in E$. In case 1, when $A = \{s\}$, any maximum cardinality matching covers all nodes in V i.e. $UV_0 = V$. Hence we compute weights $a_e = c(\delta(u))$ on the edges $e \in \delta(u)$, for all $u \in U$. Since $\cup_{u \in U} \delta(u) = E$ and $\delta(u_1) \cap \delta(u_2) = \emptyset$, we have correctly defined the weight a_e for all the edge e in E . In case 1, when $\bar{A} = \{r\}$ and $UV_0 = U$, we compute the weight a_e by the same way.

Now let us consider case 2. Let the minimum cut $[A, \bar{A}]$ contain arcs $(s, u_1), \dots, (s, u_t), (v_1, r), \dots, (v_l, r)$ and arcs (u_i, v_j) for some i and j , where $1 \leq i \leq t$ and $1 \leq j \leq l$. Then from Lemma 4.2, it implies that the nodes u_i and v_j of the arc covered by any maximum cardinality matching and the edges (u_i, v_j) are not an edge of a maximum cardinality matching in G (see the proof of Lem. 4.3). Let G_a be a graph obtained by deleting all such edges (u_i, v_j) from the graph G and let $\delta_a(w)$ denotes a star with the center w in G_a . Then there is no edge between the centers of any pair of stars $\delta_a(u)$ and $\delta_a(v)$. Hence, in the graph G_a we can correctly define $a_e = c(\delta_a(u_i))$ for all edges e of the stars $\delta_a(u_i)$ and $a_e = c(\delta_a(v_j))$ for all edges e of the stars $\delta_a(v_j)$. Since $c_{u_i v_j} \geq 0$, we can add the edge (u_i, v_j) to the set $\gamma(S_*)$ without fouling the optimality of $G_* = (S_*, \gamma(S_*))$. In this case, the set UV_0 contains the nodes that are either center of some star or not end of the edge of the minimum cut.

Now we need the following lemma.

Lemma 4.3. *If M_a is a maximum weight matching in the graph G_a with respect to the weight a_e on the edges e , then $|M_a| = |M|$, where M is a maximum cardinality matching in the graph G .*

Proof. Since $G_a = G$ for case 1, we will prove the lemma for case 2. We can assume that $c_e > 0$ for any $e \in E$. In fact, if $c_e = 0$ for some edge e , we define $c_e = \epsilon$ for an edge e , where ϵ is a sufficiently small number and ϵ may be chosen by the same way as for the linear programming problem so that the redefinition $c_e = \epsilon$ does not change an optimal solution to PMSP. First, we show that the edges (u_i, v_j) are not edges of a maximum cardinality matching in the graph G . To prove it, consider the structure of the graph G_0 (see the proof of Lem. 2.6). Since (u_i, v_j) are edges of the minimum cut, $z_e = 1$ for $e = (u_i, v_j)$. Then G_0 does not contain these edges. Hence, either the nodes u_i, v_j are the centers of some stars in G_0 or are the end nodes of some edges that are not in a maximum cardinality matching. Thus, in the graph G_a , there is at least one matching M that is a maximum cardinality matching in G . Suppose that $|M_a| < |M|$ in the graph G_a . Then there are nodes $u \in U$ and $v \in V$ not covered by the matching M_a in G . Since M_a is not a maximum matching, we can find an augmenting path π between nodes u and v . Let $(u_0, v_0), \dots, (u_k, v_k)$ denote the edges of π , where

$u = u_0$ and $v = v_k$. Then we can find new matching M_1 by inclusion and exclusion the edges of π alternately to M_a and from M_a . Since the number of edges of π is odd, the edges (u_0, v_0) and (u_k, v_k) are in M_1 . Here it follows that $|M_1| \geq |M_a| + 1$ and all the nodes $u_0, v_0, \dots, u_k, v_k$ are covered by M_1 . From the definition of the weight a_{e_i} ($e_i = (u_i, v_i)$), it follows that either $a_{e_i} = c(\delta(u_i))$ or $a_{e_i} = c(\delta(v_i))$ for $i = 0, 1, \dots, k$. Hence, the edges e_{i-1}, e_i or the edges e_i, e_{i+1} have the same weight. Since M_a and M_1 contain either the edges e_{i-1}, e_i or the edges e_i, e_{i+1} , respectively, and M_1 covers the nodes $u_0, v_0, \dots, u_k, v_k$, this implies that the weight of M_1 is greater than the weight of M_a . It contradicts that M_a is a maximum weight matching. \square

Thus, by Lemmas 4.2, 4.3 we can replace G by the bipartite graph G_a ($G_a = G$ for case 1) with the edge set $\cup_{w \in UV_0} \delta(w)$ and consider the PMSM on G_a with weights $a_e = a_w + a_l$ on the edges $e = (w, l)$ of all stars $\delta_a(w)$, where $a_w = c(\delta_a(w))$ and w is the center, $a_l = 0$ and l is a leaf of the star $\delta_a(w)$. Therefore, we derive the following algorithm to solve PMSP:

1. **Find the minimum cut $[A, \overline{A}]$ in the network G_{sr} .**
2. **If $(s, u), (v, r)$ are arcs of the cut $[A, \overline{A}]$ then delete the arc (u, v) from the graph G .**
3. **Define weight $a_e = c(\delta(w))$ on an edge e of all stars $\delta(w)$ in the graph G_a (set $G_a = G$ for case 1).**
4. **Find a maximum weight matching M_a in the graph G_a .**
5. **Find the matchable set**

$$S_* = \{w; \text{ a node } w \text{ is covered by the matching } M_a\},$$

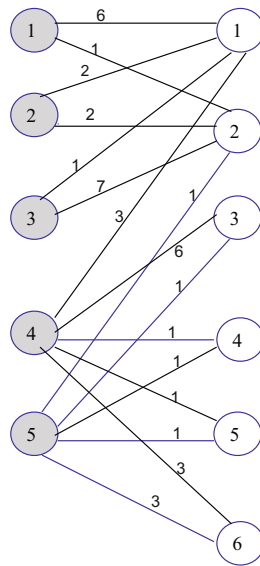
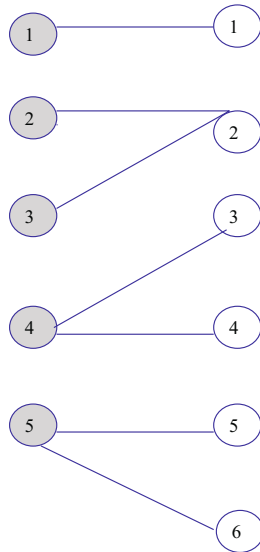
and the induced subgraph $G_* = (S_*, \gamma(S_*))$.

Consider the PMSP on the graph G in Figure 1 to illustrate the steps of the algorithm. The set U contains dark nodes and the set V contains light nodes. The weight c_e of an edge is shown near of an edge e . Figure 2 shows the graph G_0 (see Lem. 2.6). The graph G_0 contains the matching $M_0 = \{(1, 1)\}$ and the star $\delta_0(2)$ with the center $2 \in V$ and the stars $\delta_0(4), \delta_0(5)$ with centers $4, 5 \in U$. Figure 3 shows the cut $[A, \overline{A}]$ whose edges cross the curved line in the network G_{sr} . Figure 4 shows the graphs G_a and G_* . The graph G_a is obtained after deleting the edges $(4, 1)$ and $(5, 2)$ from G (see Case 2). The weights a_e are shown on the edges of the stars $\delta_a(1), \delta_a(2)$ and $\delta_a(3)$ with the centers $1, 2, 3 \in U$ and on the edges of stars $\delta_a(3), \delta_a(4), \delta_a(5), \delta_a(6)$ with the centers $3, 4, 5, 6 \in V$. In G_a , a maximum weight matching M_a contains edges $(1, 1), (3, 2), (4, 3), (5, 6)$. Hence,

$$S_* = \{\{1, 3, 4, 5\} \in U, \{1, 2, 3, 6\} \in V\}$$

which induces the graph $G_* = (S_*, \gamma(S_*))$ in Figure 4.

In conclusion, note that the AM&P algorithm finds the graph G_0 in $O(mn)$ time. Then in the network G_{sr} the minimum cut $[A, \overline{A}]$ can be found in $O(mn)$ time by using the graph G_0 . Hence the AM&P algorithm finds the cut $[A, \overline{A}]$ in

FIGURE 1. Graph G .FIGURE 2. Graph G_0 .

$O(mn)$ time. Since $|M_a| = |M|$ by Lemma 4.3, then a maximum weight matching in G_a with the weight (cost) a_e on an edge e , can be found by solving the minimum cost flow problem on the network of the underlying graph G_a with a unit capacity

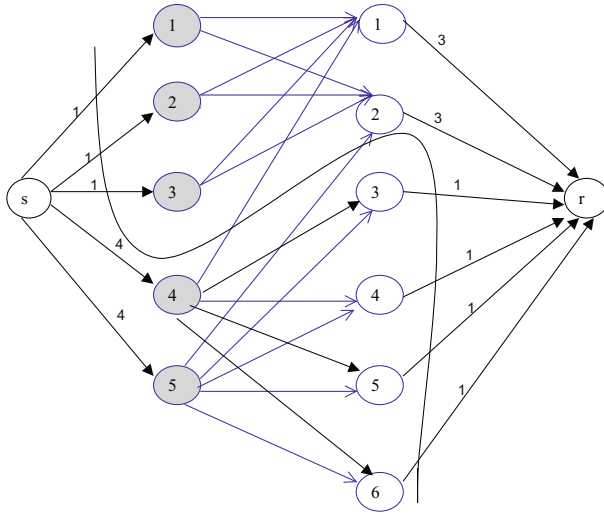


FIGURE 3. Network G_{sr} .

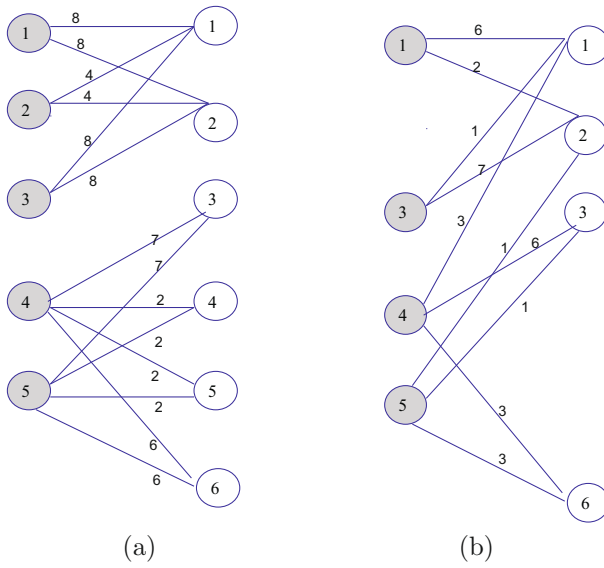


FIGURE 4. (a) Graph G_a . (b) Graph G_* .

of all arcs and with the flow value $|M| \leq n$. Thus, the primal-dual algorithm to be applied to the latter problem sends exactly unit flow from source to sink in each iteration. Therefore, the computations required to find a maximum weight matching are exactly the same as those performed for computing n shortest paths

between source and sink in the network with underlying graph G_a . Consequently, the running time of the algorithm is $O(n^3)$.

The author wish to thank an anonymous referee for his helpful remarks.

REFERENCES

- [1] R. Ahuja, T.K. Magnanti and J.B. Orlin, *Network Flows, Theory, Algorithms and Applications*. Prentice-Hall (1993).
- [2] H. Alt, N. Blum, K. Mehlhorn and M. Paul, Computing maximum cardinality matching in time $O(n^{1.5}\sqrt{m/\log|V|})$. *Infor. Process. Lett.* **37** (1991) 237–240.
- [3] E. Balas and W. Pulleyblank, The perfectly matchable subgraph polytope of a bipartite graph. *Networks* **13** (1983) 495–516.
- [4] E. Balas and W. Pulleyblank, The perfectly matchable subgraph polytope of an arbitrary graph. *Combinatorica* **9** (1989) 321–337.
- [5] M. L. Balinski, Signature methods for the assignment problem. *Oper. Res.* **33** (1985) 527–536.
- [6] D. Cornaz and A.R. Mahjoub, The Maximum Induced Bipartite Subgraph Problem with Edge Weight. *SIAM J. Discrete Math.* **3** (2007) 662–675.
- [7] W.H. Cunningham and J. Green-Krotki, A separation algorithm for matchable set polytope. *Math. Program.* **65** (1994) 139–150.
- [8] M. Grotscchel, L. Lovasz and A. Schrijver, *Geometric algorithms and combinatorial optimization*. Springer-Verlag, Berlin (1988).
- [9] F.A. Sharifov, Determination of the minimum cut using the base of an extended polymatroid. *Cybern. Syst. Anal.* **6** (1997) 856–867 (translated from *Kibernetika i Systemnyi Analis* **6** (1996) 138–152, in Russian).