An Algorithm for Multi-Attribute Diverse Matching

Saba Ahmadi¹, Faez Ahmed², John P. Dickerson¹, Mark Fuge³, Samir Khuller⁴
¹ Department of Computer Science, University of Maryland
² Department of Mechanical Engineering, Northwestern University
³ Department of Mechanical Engineering, University of Maryland
⁴ Department of Computer Science, Northwestern University
saba@cs.umd.edu, faez@northwestern.edu, john@cs.umd.edu, fuge@umd.edu, samir.khuller@northwestern.edu

Abstract

Bipartite $b$-matching, where agents on one side of a market are matched to one or more agents or items on the other, is a classical model that is used in myriad application areas such as healthcare, advertising, education, and general resource allocation. Traditionally, the primary goal of such models is to maximize a linear function of the constituent matches (e.g., linear social welfare maximization) subject to some constraints. Recent work has studied a new goal of balancing whole-match diversity and economic efficiency, where the objective is instead a monotone submodular function over the matching. Basic versions of this problem are solvable in polynomial time. In this work, we prove that the problem of simultaneously maximizing diversity along several features (e.g., country of citizenship, gender, skills) is NP-hard. To address this problem, we develop the first combinatorial algorithm that constructs provably-optimal diverse $b$-matchings in pseudo-polynomial time. We also provide a Mixed-Integer Quadratic formulation for the same problem and show that our method guarantees optimal solutions and takes less computation time for a reviewer assignment application.

1 Introduction

The bipartite matching problem occurs in many applications such as healthcare, advertising, and general resource allocation. Weighted bipartite $b$-matching is a generalization of this problem where each node on one side of the market can be matched to many items from the other side, and where edges may also have associated real-valued weights. Examples of weighted bipartite $b$-matching include assigning children to schools [Drummond et al., 2015; Kurata et al., 2017], reviewers to manuscripts [Charlin and Zemel, 2013; Liu et al., 2014], and donor organs to patients [Dickerson and Sandholm, 2015; Bertsimas et al., 2019].

Ahmed et al. [2017] introduced the notion of diverse bipartite $b$-matching, where the goal was to simultaneously maximize the “efficiency” of an assignment along with its “diversity.” For example, a firm might want to hire several highly-skilled workers, but if that firm also cares about diversity it may want to ensure that some of those hires occur across marginalized categories of employees. They proposed an objective which combined economic efficiency and diversity demonstrating that, in practice, reducing the efficiency of a matching by small amounts can often lead to significant gains in diversity across a matching. However, their formulation was limited to diversity for a single feature. It also relied on solving a general Mixed-Integer Quadratic Program (MIQP), which is flexible but computationally intractable.

In this work, we generalize the diverse matching problem and introduce matchings where each worker has multiple features (e.g., country of origin, gender) and our goal is to form diverse teams with respect to all these features. We found that the problem with a single feature, studied by Ahmed et al. [2017], can be reduced to a minimum quadratic cost maximum flow formulation and solved in polynomial time by an existing algorithm [Minox, 1986]. In contrast, we provide NP-hardness results for the general case of multiple features.

Our contributions. The paper’s main contributions follow:

- We provide the first pseudo-polynomial time algorithm for the diverse bipartite $b$-matching w.r.t. multiple features problem with class-specific weights.¹ The key insight lies in detecting negative cycles in an auxiliary graph representation, which we use to either provide incremental improvements to the incumbent diverse matching or prove that our negative-cycle-detection algorithms have found

¹That is, under conditions when the cost of assigning all items from one category to an item on the other side of the graph is the same. This holds when, e.g., one is matching academic papers to reviewers where each reviewer can specify exactly one field of expertise and the cost of assigning a paper to any of the reviewers within the same field is the same but differs across fields.

Figure 1: An illustrative example of single feature diverse matching (left) versus multi-feature diverse matching (right); here, the matching creates teams with workers from each country and gender.
a globally-optimal matching. We also provide a general MIQP formulation for this problem.

- We then extend the algorithm to the diverse bipartite b-matching problems with general edge weights, where edge weights of nodes within a category can be different.
- Lastly, we demonstrate our algorithm’s applicability to paper-reviewer matching. Our algorithm takes less time to converge to an optimal solution than the proposed MIQP approach (using a state-of-the-art commercial solver).

2 Related Work

Matching people to form diverse teams leverages the intersection of two past areas of research: the role of team diversity in collaborative work and how diversity among groups of resources is measured and used to form/match teams. Compared to related work, this paper provides a practical, high-performance method to perform diverse b-matching that can enable applications like diverse team formation or diverse resource allocation. Below we will use the example of diverse team formation (for example, in project teams within a company) to provide a concrete example to place prior work in context; however, our proposed approach is generally applicable to any diverse matching problem.

In the example of forming teams, the traditional approach is to use weighted bipartite b-matching (WBM) methods [Basu Roy et al., 2015]. These methods maximize the total weight of the matching while satisfying some constraints. However, there are two major issues with these approaches. First, it assumes that the value provided by a person in a team is always fixed and independent of who else is in the team. This assumption may not hold in many cases. A new team member may provide more added value to the team if she is added to a smaller team compared to the case if she is added to a larger team. This property of diminishing marginal utility can be mathematically captured by a family of functions called submodular functions. Second, existing approaches do not account for diversity within a team, where teams with workers from different backgrounds may be desirable. For example, different types of worker diversity have a direct impact on the success rate of tasks [Ross et al., 2010]. Likewise, firms with a higher number of employees with higher education and diversity in the types of educations have a higher likelihood of innovating [Østergaard et al., 2011] and increasing revenue for firms [Hunt et al., 2015]. In this paper, we address both these issues.

Past researchers have generally measured diversity by defining some notion of coverage—that is, a diverse set is one that covers the space of available variation. Mathematically, researchers have done so via the use of submodular functions, which encode the notion of diminishing returns [Lin and Bilmes, 2012]; that is, as one adds items to a set that are similar to previous items, one gains less utility if the existing items in the set already “cover” the characteristics added by that new item. For example, many previous diversity metrics used in the information retrieval or search communities—including Maximum Marginal Relevance (MMR) [Carbonell and Goldstein, 1998] and Determinantal Point Processes [Kulesza et al., 2012]—are instances of submodular functions. These functions can model notions of coverage, representation, and diversity [Ahmed and Fuge, 2018] and have been shown to achieve top results on common automatic document summarization benchmarks—e.g., at the Document Understanding Conference [Lin and Bilmes, 2012].

Within matching, our work is closest to that of Ahmed et al. [2017], which used a supermodular function to propose a diverse matching optimization method. Other researchers have also approached similar problems, with diversity either as an objective or as a constraint. For instance, Götz and Procaccia [2019] match migrants to localities in a way that maximizes the expected number of migrants who find employment. Benabbou et al. [2018] study the trade-off between diversity and social welfare for the Singapore housing allocation. They model the problem as an extension of the classic assignment problem, with additional diversity constraints. Lian et al. [2018] solve the assignment problem when preferences from one side over the other side are given and both sides have capacity constraints. They use order weighted averages to propose a polynomial-time algorithm which leads to high quality and more fair assignments. Agrawal et al. [2018] show that a simple iterative proportional allocation algorithm can be tuned to produce maximum matching with high entropy. Finally, Kobren et al. [2019] proposed two fairness-promoting algorithms for the paper-reviewer matching problem. They demonstrate that their algorithm achieves higher utility compared to state of the art matching algorithms that optimize for fairness only. In contrast, our goal is to develop an algorithm for finding the optimal assignment which maximizes utility as well as diversity along multiple features as an objective—along with having constraints on workload.

We define a utility function that can be tuned to balance the diversity and total weight of matching. The diversity function is inspired by the Herfindahl index [Hirschman, 1964], which is a statistical measure of concentration and commonly used in economics. We provide a new algorithm that models the problem using an auxiliary graph and uses a heuristic improvement of the negative cycle detection of Bellman-Ford by Goldberg and Radzik [1993] to find negative cycles and cancel them on a new graph to obtain an optimal solution for the original problem.

3 Preliminaries

In this section, we first define the preliminaries for a diverse matching problem, where workers are to be matched to teams and each team wants workers belonging to a diverse set of features. In our problem, we are given a set of features for the workers. Let \( F = \{ f_1, \ldots, f_{|F|}\} \) denote the feature set for the workers. An example of a feature set could be \{country of citizenship, gender\}. Each feature \( f_k \in F \) has one of the values \( F_k = \{ f_{k,1}, \ldots, f_{k,|F_k|}\} \). Let \( |F_{k,k'}| \) denote the number of workers having value \( f_{k,k'} \) for feature \( f_k \). The set of workers is denoted by \( X = \{ x_1, \ldots, x_n \} \). We wish to

\[ \text{maximize } \sum_{k} \sum_{k'} \sum_{f_{k,k'}} \sum_{x_i} u(x_i, f_{k,k'}) \text{subject to } \forall k, k', f_{k,k'}, x_i \]
form a set of teams \( \{T_1, \ldots, T_t\} \) of the workers where each team \( T_i \) has a demand of \( d_i \), specifying the number of workers that need to be assigned to it. Each worker can be assigned to exactly one team.

The diversity of an assignment is denoted by \( D \) and is equal to \( \sum_{k=1}^{|F|} \lambda_k D_k \), where \( D_k \) denotes the diversity w.r.t. feature \( F_k \), and \( \lambda_k \in \mathbb{Z}^+ \) is a constant. Let \( c_{i,k,k'} \) denote the number of workers in \( T_i \) having value \( F_k,F_{k'} \in F_k \) for feature \( F_k \). Then, \( D_k = \sum_{i=1}^t \sum_{k'=1}^{|F|} c_{i,k,k'}^2 \). To facilitate explanation, we assume throughout this paper that the country of origin is the 1st feature, therefore, the number of workers assigned to team \( T_i \) from \( j \)-th country is denoted by \( c_{i,1,j} \). The cost of assigning each worker from \( j \)-th country to team \( T_i \) is denoted by \( u_{i,j} \in \mathbb{Z}^+ \). We assume all costs are integers. The total cost of an assignment is \( TU = \sum_{i=1}^t \sum_{j=1}^{|F|} u_{i,j} \cdot c_{i,1,j} \).

Our goal is to minimize the objective function which is equal to \( \lambda \cdot D + \lambda_0 \cdot TU \), where \( \lambda \in \mathbb{Z}^{+|F|} \), and \( \lambda_0 \in \mathbb{Z}^+ \) is a constant. Next, we provide Theorem 1, which shows that this problem is NP-hard.

**Theorem 1.** Minimizing the supermodular diversity function w.r.t multiple features is NP-hard.

**Proof.** We show a reduction from the 3-COLOR problem which is as follows: given a graph \( G = (V,E) \) with \( n \) vertices, does there exist a coloring with \( n \) vertices of color \( c_1 \), \( n_2 \) vertices of color \( c_2 \), and \( n_3 \) vertices of color \( c_3 \), such that no two adjacent vertices receive the same color, and all the vertices are colored?

The reduction is as following: In 3-COLOR, assign a feature \( F_k \) to each edge \( e_k = (v_{k_1},v_{k_2}) \in E \), and a worker to each vertex. Let \( f_{k,i} \) denote the value of \( f_k \) for the worker corresponding to \( v_i \in V \). Let \( f_{k,i} = i \) if \( i \neq k_1, k_2 \). Otherwise, let \( f_{k,i} = 0 \). The goal is to form three teams \( T_1,T_2,T_3 \) with demands \( d_1=n_1, d_2=n_2, d_3=n_3 \), respectively. We assume that all the costs of assigning workers to the teams are zero, therefore the objective function is to minimize the total diversity. Consider an arbitrary edge \( e_k = (v_{k_1},v_{k_2}) \). If the endpoints of \( e_k \) belong to different teams, \( f_k \) contributes \( n_1 + n_2 + n_3 \) to the objective function since all the workers inside a team have different values for \( f_k \). Otherwise, it contributes \( n_1 + n_2 + n_3 - 2 + 2d \) since workers corresponding to \( v_{k_1},v_{k_2} \) are the only workers having the same value for \( f_k \) inside a team. If the cost of the optimal solution for the diverse matching problem is \( (n_1 + n_2 + n_3) \cdot |E| \), there does not exist a pair of workers in a team where the vertices corresponding to them are neighbours in \( G \). Otherwise if the cost of the optimal solution is more than \( (n_1 + n_2 + n_3) \cdot |E| \), the 3-COLOR instance is infeasible.

We are interested in solving this NP-hard problem. We begin by presenting two different representations of instances of the problem: one in matrix form (used for expository ease), and the other in graph form (used to build our optimal diverse matching algorithm in Section 4).

**Matrix Representation:**

An example of matrix representation with three teams and two countries and two genders is shown in Fig. 2. In this representation, each column \( V_j \) corresponds to a feature set \( v_j = \{v_{j,1}, \ldots, v_{j,|F|}\} \), where \( 1 \leq k \leq |F|, v_{j,k} \in F_k \). Each row corresponds to a team. Entry \( v_{i,j} \) shows the number of workers with feature set \( v_j \) assigned to \( T_i \). We introduce a dummy team \( T_0 \), and \( v_{0,j} \) shows the number of workers with feature set \( v_j \) who are not assigned to any team.

**Matching Representation:** In this representation, a bipartite graph \( G = (X \cup T, E) \) is given. The nodes in \( X \) correspond to the workers, and are partitioned into \( |V| \) subsets \( V_1, \ldots, V_{|V|} \), where each subset corresponds to the feature set for a column in the matrix representation. Each vertex in \( T \) corresponds to one team. The assignment of workers to teams forms a \( b \)-matching, where the degree of node \( T_i \in T \) is \( d_i \), and the degree of node \( x \in X \) is at most one.

**Local Exchange:** A local exchange happens when a group of teams decides to transfer one or more workers between each other while maintaining the total number of workers in each of them. The exchange is done in a way that the initial demands of all the teams are fulfilled. Arrows in Fig. 2 show a local exchange in a matrix representation.

In this exchange, one worker from \( T_2 \) is moved to \( T_3 \) to \( T_1 \). Two workers from \( T_1 \) are moved. One is moved from \( T_1 \) to \( T_2 \), and the other one is moved from \( T_2 \) to \( T_3 \). The set of edges of local exchange in a matrix representation is called a cycle. The source-transitions of a cycle are the cells without any input edges, and the sink-transitions are the cells without any output edges. In Fig. 2, the nodes corresponding to \( w_{3,2} \) and \( w_{1,1} \) are source-transitions and the nodes corresponding to \( w_{1,2} \) and \( w_{3,1} \) are sink-transition nodes.

Figure 4 shows the same local exchange operation using a matching representation. In this figure, the black matching shows the initial assignment, and the dotted red matching shows the assignment after the exchange operation is done.

**Gain of a local exchange:** Our goal is to minimize the

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team 0</td>
<td>( c_{0,3} = w_{0,3} + w_{0,4} )</td>
</tr>
<tr>
<td>Team 1</td>
<td>( c_{1,2} = w_{1,2} + w_{1,3} )</td>
</tr>
<tr>
<td>Team 2</td>
<td>( c_{2,2} = w_{2,2} + w_{2,3} )</td>
</tr>
<tr>
<td>Team 3</td>
<td>( c_{3,2} = w_{3,2} + w_{3,4} )</td>
</tr>
</tbody>
</table>
objective function $f$, by doing some local exchanges. To find out, we first calculate the marginal gain from a given exchange operation which is the difference between the objective values before and after a local exchange. In order to simplify this concept, we use the following definition:

**Embedding of Matrix Representation:** Consider a given matrix representation $M$, it can be embedded into a matrix $M_k$ for a fixed feature $f_k$ in the following way: all the columns in $M$ corresponding to the same value $f_{k,k'}$ of $f_k$, are combined into a single column in $M_k$. For example, embedding of the matrix representation in Figure 2 into $M_1$, $M_2$ w.r.t. the features country and gender are shown in Figures 3 and 4. Since in $M_1$, the number of people assigned from each country to each team is not changed, $\Delta(\lambda_0 \cdot TU + \lambda_1 D_1) = 0$. According to $M_2$, $\Delta(\lambda_2 D_2) = \lambda_2 ((c_{1,2,2} - 1)^2 - (c_{2,3,2})^2 + (c_{1,2,2} + 1)^2 - (c_{1,2,2} - 1)^2 - c_{1,2,1} + c_{3,2,1} + 1)^2 - c_{3,2,1})^2$.

It can be seen that the contribution of the nodes which are not source-transition or sink-transition to the gain of a local exchange is zero (all the nodes in the local exchange in Figure 3, and the node corresponding to $c_{2,2,1}$ in $M_2$). If the net gain, i.e. $\Delta(\lambda_0 \cdot TU + \lambda_1 D_1 + \lambda_2 D_2)$, is negative, then the local exchange can be considered beneficial and we can transfer the workers.

4 Negative-Cycle-Detection-based Algorithms

In this section, we explain our algorithm for finding the optimum assignment. First, we build an auxiliary graph $G'$. For each team $T_i$, there is a switch in $G'$ with $|V|$ input ports, and $|V|$ output ports, where $|V|$ is the number of columns in the matrix representation. Each port is a node in $G'$, and each switch is a directed bipartite graph, with edges going from its input ports (nodes) to its output ports. In Figure 6, each box is a switch. A dummy team $T_0$ is introduced to accommodate all unassigned workers in the matching. Inside a switch $T_i$, there is a directed edge from each input port to each output port. If the directed edge is connecting two ports such that their corresponding combinations of features do not have the same value for any features, the weight of this edge is equal to zero. Otherwise, per each feature $f_k$, that has the same value, $-2\lambda_k$ is added to the weight of this edge. The reason behind assigning these weights to the edges is to make sure in a local exchange, considering a fixed feature $f_k$, the nodes which are not a source-transition or a sink-transition w.r.t. $M_k$, have zero contribution to $\Delta(D_k)$.

For each pair of teams $T_{i_1}$ and $T_{i_2}$ where $i_1 \neq i_2$, and for each feature combination $f_j$, there is a directed edge from output port $O^{i_1}_j$ of switch $T_{i_1}$ to the input port $I^{i_2}_j$ of switch $T_{i_2}$, and weight of this edge captures the difference in the objective function when in the matrix representation a person in column $V_j$ (with feature set $v_j$) is moved from $T_{i_1}$ to $T_{i_2}$.

Each cycle in this graph is corresponding to a cycle in a matrix representation and local exchanges along them have the same gain. Figure 6 shows a cycle which is corresponding to the cycles in Figures 2 and 5.

After constructing the auxiliary graph, we run Algorithm 1. Algorithm 1 moves workers from one team to another if it detects a negative cycle.

**Algorithm 1:** Find optimal diverse $b$-matching

<table>
<thead>
<tr>
<th>Input</th>
<th>Directed weighted graph $G'$, initial feasible $b$-matching $Q$ which satisfies team demands.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>Optimal diverse $b$-matching $Q'$ which satisfies team demands.</td>
</tr>
<tr>
<td>while</td>
<td>$\exists$ a negative cycle $C \in G'$ do</td>
</tr>
<tr>
<td>// Perform a local exchange operation along $C$;</td>
<td></td>
</tr>
<tr>
<td>for $e \in C$ do</td>
<td></td>
</tr>
<tr>
<td>// Assume edge $e$ is from output port $O^{i_1}<em>j$ of team $T</em>{i_1}$ to input port $I^{i_2}<em>j$ of another team $T</em>{i_2}$;</td>
<td></td>
</tr>
<tr>
<td>// Move one worker with feature set $v_j = {f_1,k'<em>1,\cdots,f</em>{</td>
<td>F</td>
</tr>
<tr>
<td>$\forall k \in {1,\cdots,</td>
<td>F</td>
</tr>
<tr>
<td>$c_{i_1,k,k'<em>1} -= 1, c</em>{i_2,k,k'_1} += 1$;</td>
<td></td>
</tr>
<tr>
<td>Update weight of edges of $G'$ w.r.t to the new values of $c_{i_1,k,k'<em>1}$, and $c</em>{i_2,k,k'_1}$;</td>
<td></td>
</tr>
</tbody>
</table>

Algorithm 1 takes as input an initial feasible solution $Q$ as input. To find $Q$, we first find a feasible solution, which satisfies all the demand constraints. In order to find an initial feasible solution, in each iteration, consider the first subset of workers in the bipartite graph $G (V_j)$ with at least one un-assigned worker, and the first team ($T_i$) such that the number of workers assigned to it is less than its demand (In the first iteration, we start with $V_j$, $T_i$, and all the workers are un-assigned). Assign un-assigned workers from $V_j$ to $T_i$, until either demand of $T_i$ is fully satisfied, in this case, move to the next team ($i = i + 1$), or all the workers from $V_j$ are assigned, then let $j = j + 1$. Repeat this procedure until all the demand constraints are satisfied. Time complexity of this procedure is $O(|V| + t)$.

In Algorithm 1, any negative cycle detection algorithm can be used to detect negative cycles in $G'$. We use a heuristic improvement of Bellman-Ford proposed by Goldberg and Radzik [Goldberg and Radzik, 1993] in our experiments.
5 Proof of Optimality

In this section, we prove that Algorithm 1 gives the optimal solution for diverse bipartite $b$-matching problem.

Assume after the algorithm ends, the final assignment is a local optimum $P$, and the optimal solution is $P^*$. Consider the matching representations of $P$ and $P^*$. The symmetric difference of $P$ and $P^*$ ($P \odot P^*$) can be decomposed into a set of alternating cycles and paths of even length. The reason that the length of alternating paths is even is that size of both of the matchings is equal: $|P| = |P^*| = \sum_i d_i$.

Each local exchange along an alternating cycle corresponds to a cycle in the matrix representation. A local exchange along an alternating path corresponds to a cycle in the matrix representation which includes vertices from row $T_0$.

Before proving Thm. 2, we need the following definitions:

Maximal Cycle: A cycle $y$ in a matrix representation $M$ is maximal if its source-transitions (nodes with zero incoming edges) and sink-transitions (nodes with zero outgoing edges) are source-transition and sink-transition w.r.t all the edges in $M$ as well. For example, consider Figure 7. Let's call the green cycle $y_g$, the red cycle $y_r$, and the blue cycle $y_b$. The $y_g$ has two source-transitions $w_{1,1}, w_{0,3}$, and it has two sink-transitions $w_{0,1}, w_{1,3}$. Since there are no edges going out of $w_{1,3}, w_{0,1}$, and no edges going into $w_{0,3}, w_{1,1}$, $y_g$ is a maximal cycle. Cycles $y_r, y_b$ are maximal cycles as well. Therefore, $\{y_g \cup y_r \cup y_b\}$ gives a maximal cycle decomposition for $M$. However, if we consider embedding of $M$ w.r.t gender ($M_2$), then $y_r$ is not a maximal cycle anymore, and $\{y_g, y_r, y_b\}$ gives a maximal cycle decomposition w.r.t $M_2$ and $M_1$ (embedding w.r.t countries). A cycle is called all-maximal cycle if it is maximal w.r.t all the matrix representations $M_1, \cdots, M_1$. In this example, $\{y_g, y_r, y_b\}$ gives an all-maximal cycle decomposition.

Lemma 1. The set of all the edges of $P \odot P^*$ can be decomposed into a set of all-maximal cycles.

Proof. Consider an arbitrary decomposition of the edges of $P \odot P^*$ in the matrix representation into a set of cycles $\{y_1, \cdots, y_r\}$. If there exists a cycle in $P \odot P^*$ without any source-transitions and sink-transitions, it means the gain of this cycle is zero and it could be discarded. If there exists any cycle $y_k$ which is not maximal, then there exists another cycle $y_g$ which makes $y_k$ not to be maximal w.r.t some features. For example in Figure 7, $y_k$ is not maximal because of $y_g$. In this case, union $y_k$ and $y_g$, and make $y_k \cup y_g$ a single cycle in the decomposition. At the end, all the edges in $P \odot P^*$ will be decomposed into a set of all-maximal cycles. Let's call the set of all-maximal cycles $\{y_1', \cdots, y_r'\}$.

Theorem 2. Algorithm 1 finds the global optimum for the diverse $b$-matching problem.

Proof. Let $f(P)$ show the value of the objective function for the assignment $P$. $f(P^*) = f(P) < 0$ therefore:

$$f(P^*) - f(P) = \text{gain}(y_1') + \text{gain}(y_2') + \cdots + \text{gain}(y_r') < 0$$

Where $y_k' (1 \leq k \leq r')$ is the $k$th cycle in the maximal cycle decomposition, and $y_{g,k}$ is applying the local exchange of the cycle $y_k'$ at step $k$. The initial step is the assignment $P$. Since $f(P^*) - f(P) < 0$, there must be a maximal cycle $y_k'$ such that $\text{gain}(y_k') < 0$. We wish to show $\text{gain}(y_k') < 0$, which implies starting from the initial assignment $P$, a local exchange can be done with a negative gain, and $P$ is not a local optimum which is a contradiction.

Let $D(y_g), U(y_g)$ denote respectively the change in the diversity, and the change in the utility when applying a local exchange $y_g'$ in step $g$. Let $D_f(y_g) = \text{gain}(y_g') < 0$ implies $\text{gain}(y_g') < 0$, and the proof is complete.

Lemma 2. If $D_f(y_g') < 0$, then $D_f(y_g') < 0$.

Proof. Consider $y_g'$ embedded into $M_k$. There are four types of vertices in $y_g'$:

1. Vertices in the form of $w_{0,j}$ where $1 \leq j \leq |V|$. These vertices have contribution zero to both $D_f(y_g')$ and $D_f(y_g')$.
2. Vertices that are not sink-transition or source-transition, i.e. $w_{2,2}$ in Figure 2, w.r.t $M_k$. It could be seen that contribution of these nodes to both $D_f(y_g')$ and $D_f(y_g')$ is zero.
3. Sink-transitions: Consider an arbitrary sink-transition $v$ in $y_g'$. Assume the value of this node at the beginning of step $g$ is $v_g$. The contribution of $v$ to $D_f(y_g')$ is $\lambda_k((v+1)^2 - v_g^2) > 0$. Since $v$ is a sink-transition, $v_g \geq v_1$. Therefore, $\lambda_k((v+1)^2 - v_g^2) \geq \lambda_k((v_1 + 1)^2 - v_2^2)$.
4. Source-transitions: Consider an arbitrary source-transition $v$ in $y_g'$. The contribution of $v$ to $D_f(y_g')$ is $\lambda_k((v+1)^2 - v_g^2)$. Since $v$ is a source-transition, $v_1 \geq v_g$, and therefore $\lambda_k((v+1)^2 - v_g^2) \geq \lambda_k((v_1 + 1)^2 - v_2^2)$.

At the end, contribution of all the vertices to $D_f(y_g')$ is upper bounded by their contribution to $D_f(y_g')$. Therefore if $D_f(y_g') < 0$, then $D_f(y_g') < 0$.

Theorem 3. The running time of the algorithm is $O(|F| \cdot n^2 + \lambda_0 U) \cdot |V|^2 : t^2(|V| + t))$, where $U$ is the maximum cost of an initial feasible $b$-matching and $\lambda_{max} = \max\{\lambda_1, \cdots, \lambda_{|F|}\}$.

In order to prove this theorem, first we show the following lemmas hold.
Lemma 3. The number of iterations of our algorithm is at most \((\lambda_{\text{max}} \cdot |F| \cdot n^2 + \lambda_0 U)\).

Proof. The initial state of the algorithm is a feasible b-matching with cost at most \(U\). Diversity of any matching is at most \(\lambda_{\text{max}} \cdot |F| \cdot n^2\). At each iteration, we find a negative weight cycle and since all the weights are integers, its weight can be at most \(-1\). Therefore, the objective function decreases by at least 1 at each step, and since the value of the objective function is always positive, the number of iterations is at most \((\lambda_{\text{max}} \cdot |F| \cdot n^2 + \lambda_0 U)\).

Lemma 4. The complexity of each iteration of the algorithm is \(O(|V|^2 \cdot t^2(|V| + t))\).

Proof. At each iteration, we use a negative cycle detection algorithm with running time \(O(|V| \cdot |E|)\) (where \(|V|\) is the number of nodes in the auxiliary graph and \(|E|\) is the number of edges). The number of nodes in the graph is \(2|V| \cdot (t + 1)\), since there are \(t + 1\) switches in the graph and each switch has exactly \(2|V|\) ports and each port is a node in the graph. The number of edges incident on each port is \(|V| + t\). Therefore, the total number of edges is \(O(|V| \cdot t(|V| + t))\). Hence, the complexity of each iteration is \(O(|V|^2 \cdot t^2(|V| + t))\).

Combining Lemma 3 with Lemma 4, and considering \(O(|V| + t)\) time complexity for finding an initial feasible solution, yields Theorem 3.

6 Diverse Weighted Bipartite b-Matching

In this section, we extend our algorithm to solve the case where the cost of assigning workers from the same country to a team can be different. First, in each switch we put input and output ports for each worker. Inside each switch, there is a complete bipartite graph from input ports to the output ports. Consider an edge between an input port to an output port corresponding to workers \(x_i\) and \(x_j\). Per each feature \(f_k\) where \(x_i, x_j\) have the same values for \(f_k\), \(-2\lambda_k\) is added to the weight of the edge between \(x_i, x_j\).

Consider an edge from output port \(x^k_{x_i}\) of switch \(T_{i_1}\) to input port \(x^k_{x_j}\) of switch \(T_{j_1}\), where \(x_k \in V_j\). The weight of this edge is equal to the change in the objective function by moving one worker from \(V_j\) out of \(T_{i_1}\), and adding that worker to \(T_{j_1}\). The proof of the following theorem is similar to Thm. 3.

Theorem 4. The running time of the algorithm for general weights is \(O((\lambda_{\text{max}} \cdot |F| \cdot n^2 + \lambda_0 U) \cdot n^2 \cdot t^2(n + t))\), where \(U\) is the maximum cost of any feasible b-matching.

7 Experimental Validation & Discussion

To demonstrate the efficacy of the proposed method, we apply it to a dataset of reviewer paper matching. First, we find the optimal solution for multi-feature reviewer paper matching and compare it to the single feature diverse matching method. We also provide the MIQP formulation of the same problem based on literature and show how our algorithm is faster to the Gurobi based MIQP solver.

For the reviewer assignment problem, where each reviewer has multiple features, we want to match each paper with reviewers who are not only from different expertise areas (clusters), but also belong to different genders. We use the multi-aspect review assignment evaluation dataset [Karimzadehgan and Zhai, 2009], a benchmark dataset from UIUC. It contains 73 papers accepted by SIGIR 2007, and 189 prospective reviewers who had published in the main information retrieval conferences. The dataset provides 25 major topics and for each paper in the set, an expert provided 25-dimensional label on that paper based on a set of defined topics. Similarly for the 189 reviewers, a 25-dimensional expertise representation is provided.

To compare our method (Algorithm 1) with a baseline, we formulate a multi-feature MIQP variant of our problem, which is an extension of the single-feature formulation provided in [Ahmed et al., 2017] and is given by:

\[
\begin{align*}
\min \lambda_0 & \sum_{i=1}^{t} \sum_{j=1}^{t} u_{i,j} \cdot c_{i,1,j} + \sum_{k=1}^{t} \sum_{i=1}^{t} \sum_{k'=1}^{t} \lambda_k \cdot \sum_{i=1}^{t} \sum_{k'=1}^{t} \sum_{k''=1}^{t} \sum_{k'''}^{t} c_{i,k,k'} \cdot c_{i,k',k''} \\
& + \sum_{k=1}^{t} \sum_{k'=1}^{t} \sum_{k''=1}^{t} c_{i,k,k'}, \forall 1 \leq i \leq t \\
& + \sum_{k=1}^{t} c_{i,k,k'}, |F_k|, 1 \leq k \leq |F|, 1 \leq k' \leq |F_k|
\end{align*}
\]

To set up the graph for our method, we first cluster the reviewers into 5 clusters based on their topic vectors using spectral clustering. To calculate the relevance of each cluster for any paper, we take the average cosine similarity of label vectors of reviewers in that cluster and the paper. We set the constraints such that each paper matches with exactly 4 reviewers, and no reviewer is allocated more than 1 paper. To increase dataset size, we double the number of reviewers by creating a copy of each reviewer. As the original dataset lacks gender information, we added a new feature to each reviewer in this dataset by randomly adding one of two gender labels (Male or Female) to each reviewer. We set \(\lambda_0 = \lambda_1 = \lambda_2 = 1\) for our experiments.

We run the negative cycle detection algorithm, and the MIQP solver using Gurobi to find the optimum solution. On converging to the optimal solution, we find that all 73 papers receive two male reviewers and two female reviewers, which shows that the method was capable of balancing gender diversity. All papers receive reviewers from four different clusters too. If we only optimize for cluster diversity, it is possible that the gender ratio for individual paper gets skewed. When we run the same model with \(\lambda_0 = 0\) (no weight to gender diversity), we find that out of 73 papers, 12 papers receive all four reviewers of the same gender and 41 papers receive three reviewers of the same gender. Hence, only 27.3% teams of reviewers are gender balanced. However, one should note that when we do not keep gender as an objective, the resultant allocation is random and different skewness can be observed in different runs based on the initial solution.

Finally, we compare the timing performance of our algorithm with MIQP by changing the number of papers that need
to be reviewed on a Dell XPS 13 laptop with i7 processor. For MIQP, we set a maximum run time of four hours (14400 seconds) for Gurobi solver, at which we report the current best MIQP solution. Table 1 shows that for all cases with the number of papers greater than 13, MIQP does not converge within four hours, while our method finds the optimum solution in lesser time. Interestingly, MIQP current solutions are found to be the same as the optimum solution found by our method, which shows that for this application, MIQP was able to search the solution but it was not able to prove that the solution is optimum. In contrast, our method finds the solution faster as well as guarantees that it is optimum.

<table>
<thead>
<tr>
<th># Papers</th>
<th># Reviewers</th>
<th>MIQP Time (s)</th>
<th>Our Method Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
<td>378</td>
<td>24.68</td>
<td>0.18</td>
</tr>
<tr>
<td>13</td>
<td>378</td>
<td>3979.90</td>
<td>14.84</td>
</tr>
<tr>
<td>23</td>
<td>378</td>
<td>14400.00</td>
<td>122.96</td>
</tr>
<tr>
<td>33</td>
<td>378</td>
<td>14400.00</td>
<td>400.56</td>
</tr>
<tr>
<td>43</td>
<td>378</td>
<td>14400.00</td>
<td>825.95</td>
</tr>
<tr>
<td>53</td>
<td>378</td>
<td>14400.00</td>
<td>2837.15</td>
</tr>
<tr>
<td>63</td>
<td>378</td>
<td>14400.00</td>
<td>5453.58</td>
</tr>
<tr>
<td>73</td>
<td>378</td>
<td>14400.00</td>
<td>11040.55</td>
</tr>
</tbody>
</table>

Table 1: Comparison of MIQP and our method for UIUC reviewer dataset with each paper needing 4 reviewers.

8 Conclusion & Future Research

In this paper, we proposed the first pseudo-polynomial time algorithms for multi-feature diverse weighted bipartite b-matching—a problem that we also showed is NP-hard. We propose an algorithm that not only guarantees an optimal solution but also converges faster than a proposed approach using a black-box industrial MIQP solver. We demonstrated our results on a dataset for paper reviewer matching. Future work could explore the extension of this method to online diverse matching [Dickerson et al., 2019], where vertices arrive sequentially and must match immediately; this has direct application in advertising, where one could balance notions of reach, frequency, and immediate monetary return. Exploring connections to fairness in machine learning [Grgić-Hlača et al., 2018] and hiring [Schumann et al., 2019] by way of diversity are also of immediate interest.

References


[Karimzadehgan and Zhai, 2009] Maryam Karimzadehgan and ChengXiang Zhai. Constrained multi-aspect expertise matching—a problem that we also showed is NP-hard. We propose an algorithm that not only guarantees an optimal solution but also converges faster than a proposed approach using a black-box industrial MIQP solver. We demonstrated our results on a dataset for paper reviewer matching. Future work could explore the extension of this method to online diverse matching [Dickerson et al., 2019], where vertices arrive sequentially and must match immediately; this has direct application in advertising, where one could balance notions of reach, frequency, and immediate monetary return. Exploring connections to fairness in machine learning [Grgić-Hlača et al., 2018] and hiring [Schumann et al., 2019] by way of diversity are also of immediate interest.

References


