On Tree-Constrained Matchings and Generalizations*

Stefan Canzar[1], Khaled Elbassioni[2], Gunnar W. Klau[1], and Julián Mestre[3]

[1] Centrum Wiskunde & Informatica, Life Sciences Group, Science Park 123, 1098 XG, Amsterdam, the Netherlands, {stefan.canzar,gunnar.klau@cwi.nl}

[2] Max-Planck-Institut für Informatik, Algorithms and Complexity Dept., Saarbrücken, Germany, elbassio@mpi-inf.mpg.de

[3] School of Information Technologies, The University of Sydney, Australia

Abstract. We consider the following Tree-Constrained Bipartite Matching problem: Given two rooted trees $T_1 = (V_1, E_1)$, $T_2 = (V_2, E_2)$ and a weight function $w : V_1 \times V_2 \mapsto \mathbb{R}_+$, find a maximum weight matching $\mathcal{M}$ between nodes of the two trees, such that none of the matched nodes is an ancestor of another matched node in either of the trees. This generalization of the classical bipartite matching problem appears, for example, in the computational analysis of live cell video data. We show that the problem is $\mathcal{APX}$-hard and thus, unless $\mathcal{P} = \mathcal{NP}$, disprove a previous claim that it is solvable in polynomial time. Furthermore, we give a 2-approximation algorithm based on a combination of the local ratio technique and a careful use of the structure of basic feasible solutions of a natural LP-relaxation, which we also show to have an integrality gap of $2 - o(1)$. In the second part of the paper, we consider a natural generalization of the problem, where trees are replaced by partially ordered sets (posets). We show that the local ratio technique gives a $2k\rho$-approximation for the k-dimensional matching generalization of the problem, in which the maximum number of incomparable elements below (or above) any given element in each poset is bounded by ρ. We finally give an almost matching integrality gap example, and an inapproximability result showing that the dependence on ρ is most likely unavoidable.

Key words: k-partite matching, rooted trees, approximation algorithms, local ratio technique, inapproximability, computational biology

1 Introduction

This paper contains both approximation and hardness results for the Tree-Constrained Bipartite Matching (TCBM) problem, a natural generalization of the classical maximum matching problem in bipartite graphs. The input of TCBM consists of a weighted bipartite graph $G = (V_1, V_2, E)$ and two rooted trees T_1 and T_2. The vertex set of T_i is V_i for $i = 1, 2$. The objective is to find a maximum weight matching such that the matched vertices in each tree are not comparable; that is, if $u, v \in V_i$ are matched then u cannot be v's ancestor or vice-versa. Figure 1 illustrates the definition.

TCBM arises naturally in the computational analysis of live cell video data. Studying cell motility using live cell video data helps understand important biological processes, such as tissue repair, the analysis of drug performance, and immune system responses. Segmentation based methods for cell tracking typically follow a two stage approach (see [15] for a survey): The goal of the first *detection* step is to identify individual cells in each frame of the video independently. In a second step, the linkage of consecutive frames, and thus the *tracking* of a cell, is achieved by assigning cells identified in one frame to cells identified in the next frame. However, limited contrast and noise in the video sequence often leads to *over-segmentation* in the first stage: a single cell is comprised of several segments. A major challenge in this application domain is therefore the ability to distinguish biological cell division from over-segmentation.

* This technical report is the full version of [5], which appeared in the proceedings of the 38th International Colloquium on Automata, Languages and Programming (ICALP 2011).

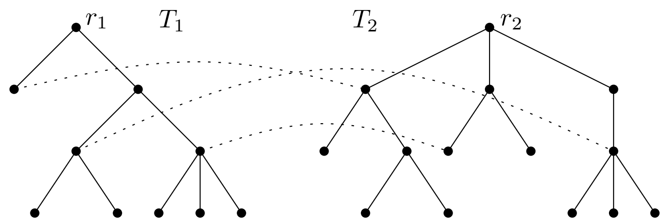


Fig. 1. Example of a feasible tree-constrained bipartite matching. For each matched pair of vertices, indicated by dotted lines, neither of their descendants are matched.

Mosig et al. [11] and Xiao et al. [13] address this problem by proposing a novel approach for the *linkage* stage. As opposed to previous methods, they match *sets of segments* between neighboring frames rather than singletons, where the segment sets correspond to the nodes of an agglomerative hierarchical clustering tree. A subsequent bipartite matching between the nodes of the clustering trees corresponding to neighboring frames integrates the identification of a cell as a set of segments and the tracking of the cell between two different video frames. Since segment sets representing different cells in the same frame must be disjoint, no two nodes on any root-to-leaf path can be matched at the same time, leading to an instance of TCBM . To assess the quality of such a tree-constrained matching, Mosig et al. consider the relative overlap of the convex hulls of matched segment sets. This *cosegmentation* via TCBM promises to be usefull also in other bioimaging applications, for example in protein-colocalization studies [13].

To track cells not only between two consecutive frames but across a whole video sequence, the bipartite graphs have to be concatenated to a *cell connection graph*, as introduced in [14]. In [11] this is done by solving a standard maximum weight bipartite matching problem for each frame i, which is at the intersection between the tree-constrained alignment of frames $i-1$ and i, and the alignment of frames i and $i+1$. Concluding, the authors mention post-processing the cell connection graph as a promising improvement. Therefore, the generalization of bipartite tree-constrained matching to a tree-constrained matching in a k-partite k-uniform hypergraph is an important problem for this application. By linking more than two frames at the same time, over-segmentation and cell-division can be distinguished by taking into account the cell behavior over a larger time-scale.

Another natural generalization of the problem is obtained by replacing trees by partially ordered sets (posets), because they permit the representation of alternative clustering hierarchies. For example, various meaningful distance measures between (sets of) segments could make it necessary to assign them to multiple parent clusters. In particular, noise in the video data may make it difficult to determine a unique tree.

Mosig et al. [11] present a linear programming formulation for TCBM and claim that the constraint matrix is totally unimodular, which would imply that the problem is solvable in polynomial time [12]. We disprove this statement by showing an instance with a fractional vertex and proving that the problem is in fact $\mathcal{NP}$-hard and even hard to approximate within a constant. Thus, conditional on $\mathcal{P} \neq \mathcal{NP}$, there is no polynomial time algorithm for our problem.

TCBM and its generalization to k trees are special cases of the maximum weighted independent set (MWIS) problem on 2-interval graphs and k-interval graphs, respectively. The connection is given by ordering the leaves of the trees by depth-first search and identifying each node with the interval of leaves below it. In fact, TCBM captures precisely the subclass of 2-*union* graphs (the first interval of a 2-interval cannot intersect the second interval of another 2-interval) where the two interval families are *laminar* (any two intervals are either disjoint or one is nested in the other). In [3] the fractional local ratio technique was developed and applied to MWIS in k-interval graphs to get a $2k$-approximation algorithm. This result immediately implies a 4-approximation for TCBM.

1.1 Our Results

In this paper, we give a 2-approximation algorithm for TCBM, improving upon the 4-approximation that follows from the work of Bar-Yehuda et al. [3]. Our method is based on a combination of the local ratio technique and a careful use of the structure of basic feasible solutions of a natural LP-relaxation. In Section 2.1 we show a 3-approximation based on fractional local ratio and prove that this is the best guarantee the fractional local ratio technique alone can deliver when rounding one coordinate at a time. The main difference between our approach and that of Bar-Yehuda et al. [3] is that we round basic feasible solutions. This allows us to exploit their structure in the analysis in order to get better approximation guarantees. In Section 2.2, we show how to get a 2-approximation and give an instance for which our LP-relaxation has an integrality gap of $2-o(1)$. In Section 2.3, we show that the problem is $\mathcal{APX}$-hard. Our results imply that the MWIS problem on 2-union graphs in which both families of intervals are *laminar*, is still $\mathcal{APX}$-hard, but can be approximated within a factor of 2.

In Section 3, we consider the k-dimensional generalization of the problem to posets. In this case, the natural LP-relaxation has an exponential number of constraints, but admits an alternative linear-size LP-formulation. Even though the result of Bar-Yehuda et al. [3] does not apply directly to the poset case, we show that the fractional local ratio technique yields a $2\sum_{i=1}^{k}\rho(\mathcal{P}_i)$-approximation here, where $\rho(\mathcal{P}_i)$ is the maximum number of incomparable elements below (or above) any given element in poset $\mathcal{P}_i$. We also give an example which shows that the integrality gap of the LP-relaxation is tight within almost a factor of 2. Finally, Section 3.2 gives a reduction from *Maximum Label Cover* showing that the 2-dimensional matching problem with poset constraints is hard to approximate within a factor of $2^{\log^{1-\epsilon}\rho}$, for any $\epsilon > 0$, where $\rho = \max\{\rho(\mathcal{P}_1), \rho(\mathcal{P}_2)\}$, unless $\mathcal{NP} \subseteq DTIME(n^{\mathrm{polylog}\, n})$. Note that the k-dimensional version of TCBM includes as a special case the k-dimensional matching problem, and hence [8] is NP-hard to approximate within a factor of $O(k/\log k)$.

We conclude in Section 5 with some further generlizations of the above problems where the tree or poset constraints are replaced by independent set constraints in graphs with certain properties, such as perfect graphs, or graphs with low inductive independence numbers.

2 Matching Trees

In this section we focus on the basic TCBM problem, formally defined as follows:

Definition 1 (Tree-constrained bipartite matching problem, TCBM). *Given two rooted trees $T_1 = (V_1, E_1)$, $T_2 = (V_2, E_2)$ with roots $r_1 \in V_1$ and $r_2 \in V_2$, and a weight function $w : V_1 \times V_2 \mapsto \mathbb{R}_+$, find a maximum weight matching $\mathcal{M}$ in the complete bipartite graph $G = (V_1, V_2, E)$ with edge weights induced by w, such that $(u, v) \in \mathcal{M}$ implies $(u', v') \notin \mathcal{M}$, if u' is a descendant of u or v' is a descendant of v.*

Consider an instance (T_1, T_2, w) of TCBM with $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$. Let $r_1 \in V_1$ and $r_2 \in V_2$ denote the roots and $\mathcal{L}_1 \subset V_1$ and $\mathcal{L}_2 \subset V_2$ denote the set of leaves of the trees, respectively. For two vertices p and q denote by $[p, q]$ the path in T_1 (or T_2) between p and q. For each $p_1 \in V_1$ and $p_2 \in V_2$ let $x_{p_1,p_2} \in \{0, 1\}$ be a variable which takes value 1 if and only if p_1 is matched to p_2, i.e. $(p_1, p_2) \in \mathcal{M}$. Consider the following LP-relaxation of the problem.

$$\max \sum_{p_1\in V_1, p_2\in V_2} w_{p_1,p_2} x_{p_1,p_2} \tag{P}$$

$$\text{s.t.} \sum_{p_1\in[r_1,\ell], p_2\in V_2} x_{p_1,p_2} \le 1 \qquad \text{for all } \ell \in \mathcal{L}_1 \tag{1}$$

$$\sum_{p_1\in V_1, p_2\in[r_2,\ell]} x_{p_1,p_2} \le 1 \qquad \text{for all } \ell \in \mathcal{L}_2 \tag{2}$$

$$x_{p_1,p_2} \ge 0 \qquad \text{for all } p_1 \in V_1,\ p_2 \in V_2\ .$$

For the purpose of the analysis of the algorithm we will consider a more general LP formulation $(P_{\mathbf{b}})$, in which the right hand sides of constraints (1) and (2) are replaced by some b_l, $0 \leq b_\ell \leq 1$, for all $\ell \in \mathcal{L}_1 \cup \mathcal{L}_2$. We will use $x(F)$ to denote $\sum_{e \in F} x_e$ for a subset of the edges $F \subseteq E$ in the complete bipartite graph G and $\delta(p)$ to denote the set of edges in E incident to a vertex p.

Overview of the technique. Our main tool will be the *fractional local ratio technique*, applied (recursively) to an optimal *basic* feasible solution (BFS) x^* of the above LP, where the base case of the recursion involves the computation of maximum weight independent sets in interval graphs. As long as there is a pair $(p,q) \in V_1 \times V_2$ with "small fractional ratio", that is the total contribution $\sum_{(p',q')} x^*_{p'q'}$, over all pairs (p',q') that are in conflict with (p,q), is at most 2, we can take this pair (p,q) into the solution and recurse on a new instance with reduced weights. Otherwise (if there is no such pair), we can use a structural result about the BFS's (Lemmas 1 and 3) to reduce the problem to computing maximal independent sets in interval graphs. In the next subsection, we prove this structural result, and show the fractional local ratio approach alone can give a 3-approximation, but not more. We then extend this to a 2-approximation in Section 2.2.

2.1 A 3-Approximation by Fractional Local Ratio

For a feasible fractional solution $\mathbf{x}$ to linear program $(P_{\mathbf{b}})$ and $i \in \{1,2\}$ we denote by $\mathcal{L}_i(\mathbf{x})$ the set of nodes $\ell \in V_i$ with $x(\delta(\ell)) > 0$ and $x(\delta(p)) = 0$ for all descendants p of ℓ in T_i.

Lemma 1. *For any basic feasible solution $\mathbf{x}$ to linear program $(P_{\mathbf{b}})$ one of the following holds:*

(a) there exist nodes $\ell_1 \in \mathcal{L}_1(\mathbf{x})$ and $\ell_2 \in \mathcal{L}_2(\mathbf{x})$ such that $x_{\ell_1,\ell_2} > 0$, or
(b) for every $x_{p_1,p_2} > 0$ either $p_1 \in \mathcal{L}_1(\mathbf{x})$ and $x_{p_1,p'_2} = 0$, for all $p'_2 \neq p_2$, or $p_2 \in \mathcal{L}_2(\mathbf{x})$ and $x_{p'_1,p_2} = 0$, for all $p'_1 \neq p_1$.

Proof. Assume (a) does not hold, i.e. $x_{\ell_1,\ell_2} = 0$ for all $\ell_1 \in \mathcal{L}_1(\mathbf{x})$ and $\ell_2 \in \mathcal{L}_2(\mathbf{x})$. For an arbitrary node $p_1 \in V_1$, any two constraints (1) for leaves in the subtree rooted at p_1 linearly depend on the set of non-negativity constraints for variables $x_{p'_1,p_2}$, with $p_2 \in V_2$ and p'_1 being a descendant of p_1. Thus, every $\ell_1 \in \mathcal{L}_1(\mathbf{x})$ implies at most one linear independent tight constraint (1) for one of the leaves in its subtree. Since a symmetric argument applies to nodes $\ell_2 \in \mathcal{L}_2(\mathbf{x})$, the number of linearly independent tight constraints (1), (2) for a given basic feasible solution $\mathbf{x}$ is at most $|\mathcal{L}_1(\mathbf{x})| + |\mathcal{L}_2(\mathbf{x})|$.

On the one hand, since $x(\delta(\ell)) > 0$ for all $\ell \in \mathcal{L}_1(\mathbf{x}) \cup \mathcal{L}_2(\mathbf{x})$ and since no positive edge between two leaves exists, for every node in $\mathcal{L}_1(\mathbf{x}) \cup \mathcal{L}_2(\mathbf{x})$ there is at least one distinct non-zero edge incident to it. On the other hand, in a basic feasible solution the number of non-zero variables is at most the number of linearly independent tight constraints (1), (2), which in turn is at most $|\mathcal{L}_1(\mathbf{x})| + |\mathcal{L}_2(\mathbf{x})|$. Therefore, exactly one distinct non-zero edge is incident to every node in $\mathcal{L}_1(\mathbf{x}) \cup \mathcal{L}_2(\mathbf{x})$ and all other edges must be 0. □

Based on this property of basic feasible solutions to $(P_{\mathbf{b}})$ we next show that there always exists an edge with local ratio at most 3. For this, let $N[(p,q)]$ be the set of edges $(p',q') \in E$ that are in conflict with edge (p,q) (including (p,q) itself), i.e.,

$$N[(p,q)] = \{(p',q') \in E \mid p' \in [r_1,p] \vee p \in [r_1,p'] \vee q' \in [r_2,q] \vee q \in [r_2,q']\} \ .$$

Lemma 2. *Let $\mathbf{x}$ be a basic feasible solution to $(P_{\mathbf{b}})$. There exists an edge $(p,q) \in E$ with $x_{p,q} > 0$ such that $\sum_{(p',q') \in N[(p,q)]} x_{p',q'} \leq 3$.*

Proof. Assume there is an edge (p,q) with $x_{p,q} > 0$ such that $p \in \mathcal{L}_1(\mathbf{x})$ and $q \in \mathcal{L}_2(\mathbf{x})$. By definition, $N[(p,q)] = F_1 \cup F_2$, where $F_1 = \{(p',q') \in E \mid p' \in [r_1,p]\}$ and $F_2 = \{(p',q') \in E \mid q' \in [r_2,q]\}$. $x(F_1)$ and $x(F_2)$ are bounded by one through constraints (1), respectively (2), and thus the claimed inequality holds.

Therefore assume no such edge exists, i.e. condition (a) in Lemma 1 is not satisfied. By the same lemma we can then partition the set of non-zero edges into sets F'_1 and F'_2, which contain edges

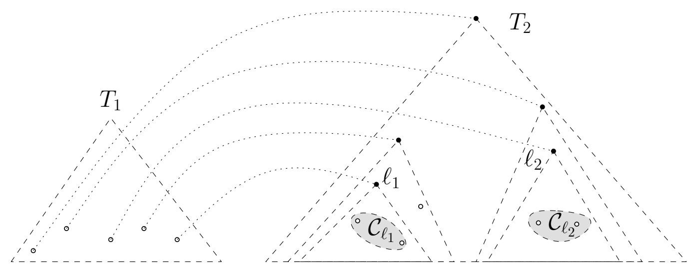


Fig. 2. Illustration of the partition of a subset of the nodes in $\mathcal{L}_2(\mathbf{x})$ as introduced in the proof of Lemma 2. In this example, $\mathcal{L}_2(\mathbf{x}') = \{\ell_1, \ell_2\}$. Only edges in F_1' are shown (dotted lines).

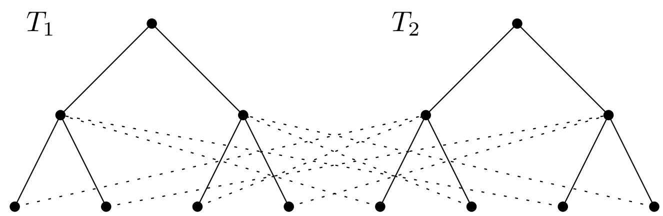


Fig. 3. Tight example for the fractional local ratio of an edge, $k = 3$.

with one endpoint in $\mathcal{L}_1(\mathbf{x})$, respectively one endpoint in $\mathcal{L}_2(\mathbf{x})$. W.l.o.g assume $x(F_1') \geq x(F_2')$. Let $x'_{p_1,p_2} = x_{p_1,p_2}$ for $(p_1, p_2) \in F_1'$ and $x'_{p_1,p_2} = 0$ for $(p_1, p_2) \in F_2'$. The nodes in $\mathcal{L}_2(\mathbf{x}')$ define a partition of a subset of the nodes in $\mathcal{L}_2(\mathbf{x})$ into sets $\mathcal{C}_\ell$, $\ell \in \mathcal{L}_2(\mathbf{x}')$, where $\mathcal{C}_\ell$ contains all descendants of ℓ in $\mathcal{L}_2(\mathbf{x})$, i.e. $\mathcal{C}_\ell = \{\ell' \in \mathcal{L}_2(\mathbf{x}) \mid \ell \in [r_2, \ell']\}$. See Fig. 2 for an illustration. We want to show that there exists a $\mathcal{C}_\ell$ such that $\sum_{\ell' \in \mathcal{C}_\ell} x(\delta(\ell')) \leq 1$. Assume $\sum_{\ell' \in \mathcal{C}_\ell} x(\delta(\ell')) > 1$ for all $\ell \in \mathcal{L}_2(\mathbf{x}')$.

$$x(F_2') \geq \sum_{\ell \in \mathcal{L}_2(\mathbf{x}')} \sum_{\ell' \in \mathcal{C}_\ell} x(\delta(\ell')) > \sum_{\ell \in \mathcal{L}_2(\mathbf{x}')} 1 \geq \sum_{\ell \in \mathcal{L}_2(\mathbf{x}')} \sum_{q \in [r_2, \ell]} x(\delta(q)) \geq x(F_1').$$

The second inequality is due to constraint (2) and the last inequality directly follows from the definition of $\mathcal{L}_2(\mathbf{x}')$. Since this contradicts our initial assumption, there must be a $(p, q) \in F_1'$ such that $\sum_{e \in F_2''} x_e \leq 1$, where $F_2'' = \{(p', q') \in F_2' \mid q' \neq q \wedge q \in [r_2, q']\}$. Since $p \in \mathcal{L}_1(\mathbf{x})$, $N[(p, q)] = F_1 \cup F_2 \cup F_2''$, for F_1 and F_2 as defined above, and the lemma follows. □

The following lemma shows that we can apply Lemma 2 inductively in a local ratio framework.

Lemma 3. *Let* $\mathbf{x}$ *be a basic feasible solution to* $(P_{\mathbf{b}})$. *For an edge* (p, q) *with* $x_{p,q} > 0$, *let* $\mathbf{b}'$ *be such that* $b'_\ell = b_\ell - x_{p,q}$ *if* $p \in [r_1, \ell]$, *respectively* $q \in [r_2, \ell]$, *and* $b'_\ell = b_\ell$ *otherwise. Then* $\mathbf{x}'$ *with* $x'_{p,q} = 0$ *and* $x'_{e'} = x_{e'}$ *for* $e' \neq (p, q)$, *is a basic feasible solution to* $(P_{\mathbf{b}'})$.

Proof. Since the right hand side of constraints in $(P_{\mathbf{b}'})$ for all root to leaf paths through p, respectively q, are decreased by exactly $x_{p,q}$, $\mathbf{x}'$ is still feasible for $(P_{\mathbf{b}'})$. Suppose that $\mathbf{x}'$ can be expressed as a convex combination of two feasible solutions $\mathbf{y}$ and $\mathbf{z}$ (of $(P_{\mathbf{b}'})$). Then setting $y_{p,q} = x_{p,q}$ and $z_{p,q} = x_{p,q}$ would yield a convex combination of $\mathbf{x}$ (in $(P_{\mathbf{b}})$), a contradiction. Thus, $\mathbf{x}'$ must be a basic feasible solution to $(P_{\mathbf{b}'})$. □

Now by applying the fractional local ratio technique of [3] we immediately obtain the following result.

Theorem 1. *There is a 3-approximation algorithm for TCBM.*

We next give an example instance that shows that using the fractional local ratio method, our approximation guarantee is tight.

Algorithm 1 Tree-Matching$(F, \mathbf{w})$

Require: A BFS $\mathbf{x}$ to $(P_{\mathbf{b}})$, $0 \le \mathbf{b} \le 1$, edge set F, and weights on the edges $\mathbf{w}$.
1: **if** $F = \emptyset$ **then**
2:    **return** $\emptyset$
3: Define $F_0 = \{e \in F \mid w_e \le 0\}$
4: **if** $F_0 \neq \emptyset$ **then**
5:    **return** Tree-Matching$(F \setminus F_0, w)$
6: **if** $\forall e \in F : x(N[e]) > 2$ **then**
7:    $\mathcal{I}_1 \leftarrow$ MWIS in IG(T_1) w.r.t. weights $\bar{w}^1_p = \max_{(p,q)\in F} w_{p,q}$
8:    $\mathcal{I}_2 \leftarrow$ MWIS in IG(T_2) w.r.t. weights $\bar{w}^2_q = \max_{(p,q)\in F} w_{p,q}$
9:    **if** $\bar{w}^1(\mathcal{I}_1) \ge \bar{w}^2(\mathcal{I}_2)$ **then**
10:       **return** $\bigcup_{p\in\mathcal{I}_1}\{\text{argmax}_{(p,q)\in F}\, w_{p,q}\}$
11:    **else**
12:       **return** $\bigcup_{q\in\mathcal{I}_2}\{\text{argmax}_{(p,q)\in F}\, w_{p,q}\}$
13: **else**
14:    Let $e' \in F$ be s.t. $x(N[e']) \le 2$
15:    Decompose $\mathbf{w}$ by $\mathbf{w} = \mathbf{w}^1 + \mathbf{w}^2$ where $w^1_e := \begin{cases} w_{e'} & \text{if } e \in N[e'], \\ 0 & \text{otherwise.} \end{cases}$
16:    $\mathcal{M}' \leftarrow$ Tree-Matching$(F, \mathbf{w}^2)$
17:    **if** $N[e'] \cap \mathcal{M}' = \emptyset$ **then**
18:       **return** $\mathcal{M} = \mathcal{M}' \cup \{e'\}$
19:    **else**
20:       **return** $\mathcal{M} = \mathcal{M}'$

Lemma 4. *There exists an instance of TCBM such that the optimal solution to* (P) *is unique and the fractional local ratio of every edge is at least* $3 - o(1)$*.*

Proof. Let T_1 and T_2 be trees of height two (i.e., they have three levels) where each internal node has $k-1$ children. The edges E connecting the nodes of T_1 and T_2 are as follows. Let x be a leaf of T_1. If x is the ith child of its *parent* in T_1, we connect x with the ith child of the *root* of T_2. We connect in a similar fashion the leaves of T_2 with the children of the root of T_1. All edges have a weight of 1. Figure 3 illustrates the construction for $k = 3$.

It can be verified that the optimal fractional solution must set the value of every edge to $\frac{1}{k}$. Let (u, v) be an edge where u is a leaf and v is an internal node. Notice that (u, v) is in conflict with $k-1$ edges incident on children of v, $k-2$ edges incident on v and $k-1$ edges incident on u's parent. Since each edge carries a fractional contribution of $\frac{1}{k}$ their combined fractional value is $3 - \frac{4}{k}$. □

2.2 A 2-Approximation

The algorithm requires a basic feasible solution $\mathbf{x}$ to (P) and is initially called with an edge set F, in which all edges e with $x_e = 0$ have been removed. The idea is to recurse in a local ratio manner as long as we can find an edge with local ratio at most two. If this is not possible anymore, we exploit the specific structure of basic feasible solutions by computing *maximum weight independent sets* (MWIS) in the interval graphs IG(T_1) and IG(T_2) induced by the two trees: For $i \in \{1, 2\}$, IG(T_i) is the interval graph obtained by ordering the leaves of T_i by depth-first search and identifying each node of T_i with the interval of leaves below it. As usual, we define the maximum (in lines 7-8) over an empty set to be 0. It is not difficult to see that the matching $\mathcal{M}$ returned by the algorithm is feasible for (P). It remains to assess the quality of this solution.

Theorem 2. *Let* $\mathbf{x}$ *be a basic feasible solution to linear program* $(P_{\mathbf{b}})$*. The matching* $\mathcal{M}$ *returned by Algorithm 1 satisfies* $w(\mathcal{M}) \ge \frac{1}{2} \cdot \mathbf{w} \cdot \mathbf{x}$*.*

Proof. The proof is by induction on the number of edges having positive weight. In the base case either there is no edge with positive weight (lines 1-2) or no edge e' in line 14 exists. In the former case, the induction hypothesis clearly holds. In the latter case, according to Lemma 1 the non-zero edges can be partitioned into sets F_1 and F_2, containing edges with one endpoint in $\mathcal{L}_1(\mathbf{x})$, respectively one endpoint in $\mathcal{L}_2(\mathbf{x})$. For $i \in \{1,2\}$, let

$$\mathcal{P}_{IG}(T_i) := \left\{ \mathbf{y} \in \mathbb{R}_+^{V_i} \;:\; \sum_{p \in V_i : p \in [r_i, \ell]} y_p \leq 1, \quad \text{for all } \ell \in \mathcal{L}_i \right\}$$

be the fractional independent set polytope in the interval graph represented by tree T_i. It is well-known (see e.g. [7]) that $\mathcal{P}_{IG}(T_i)$ is integral. Given the basic feasible solution $\mathbf{x}$ of $(P_\mathbf{b})$, define $\mathbf{y}^i \in \mathbb{R}^{V_i}$, for $i \in \{1,2\}$, as follows: $y^i_p = \sum_{q:(p,q)} x_{p,q}$, for $p \in V_i$. The feasibility of $\mathbf{x}$ to $(P_\mathbf{b})$ implies that $\mathbf{y}^i \in \mathcal{P}_{IG}(T_i)$, for $i \in \{1,2\}$. Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be the independent sets computed in steps 7 and 8 of the algorithm, and $\mathcal{M}'$ be the matching computed in step 10 or 12. Then

$$\begin{aligned}
\mathbf{w} \cdot \mathbf{x} &= \sum_{(p,q) \in F_1} w_{p,q} x_{p,q} + \sum_{(p,q) \in F_2} w_{p,q} x_{p,q} \\
&\leq \sum_{p \in V_1} \bar{w}^1_p \sum_{(p,q) \in F_1} x_{p,q} + \sum_{q \in V_2} \bar{w}^2_q \sum_{(p,q) \in F_2} x_{p,q} && (3) \\
&= \sum_{p \in V_1} \bar{w}^1_p y_p + \sum_{q \in V_2} \bar{w}^2_q y_q \leq \max_{\mathbf{y}' \in \mathcal{P}_{IG}(T_1)} \bar{\mathbf{w}}^1 \cdot \mathbf{y}' + \max_{\mathbf{y}' \in \mathcal{P}_{IG}(T_2)} \bar{\mathbf{w}}^2 \cdot \mathbf{y}' && (4) \\
&= \bar{w}(\mathcal{I}_1) + \bar{w}(\mathcal{I}_2) \leq 2 \cdot \max\{\bar{w}(\mathcal{I}_1), \bar{w}(\mathcal{I}_2)\} = 2 \cdot w(\mathcal{M}'). && (5)
\end{aligned}$$

Inequality (3) follows from the definition of the weights $\bar{\mathbf{w}}^1$ and $\bar{\mathbf{w}}^2$ in lines 7, 8; inequalities (4) and (5) follow respectively from the fact that $\mathbf{y}^i \in \mathcal{P}_{IG}(T_i)$, and the integrality of $\mathcal{P}_{IG}(T_i)$, for $i \in \{1,2\}$; and the last equality is due to the choice the algorithm makes in line 9. Note that the matchings constructed in lines 10 respectively 12 are feasible solutions to the TCBM problem. Indeed, due to the structure of a basic feasible solution (Lemma 1 which will also continue to hold inductively by Lemma 3), the edges that induce an independent set in one tree end in leaves of the second tree and therefore do not conflict.

We next prove the inductive step (the rest of the argument is the same as in [2]; we include it for completeness). If F_0 is non-empty in step 4, extending $\mathbf{w}$ in the induction hypothesis by the non-positive components that were deleted in line 5 cannot make the inequality invalid. Let $\mathbf{y}'$ and $\mathbf{y}$ be the characteristic vectors of matchings $\mathcal{M}'$ and $\mathcal{M}$, obtained in lines 16 and lines 18-20, respectively. Let e' be the edge chosen in line 14. By the decomposition of $\mathbf{w}$ in line 15, $\mathbf{w}^2$ implies a smaller number of edges with positive weight than $\mathbf{w}$. By the induction hypothesis, $\mathbf{w}^2 \cdot \mathbf{y}' \geq \frac{1}{2} \cdot \mathbf{w}^2 \cdot \mathbf{x}$. From $w^2_{e'} = 0$ it also follows that $\mathbf{w}^2 \cdot \mathbf{y} \geq \frac{1}{2} \cdot \mathbf{w}^2 \cdot \mathbf{x}$. Since at least one edge in $N[e']$ is in $\mathcal{M}$ and $x(N[e']) \leq 2$ (line 14), it also holds that $\mathbf{w}^1 \cdot \mathbf{y} \geq \frac{1}{2} \cdot \mathbf{w}^1 \cdot \mathbf{x}$. The claim follows. □

We conclude this section by giving an example showing that the integrality gap of (P) matches the approximation factor attained by our algorithm.

Lemma 5. *The integrality gap of* (P) *is* $2 - o(1)$.

Proof. Our bad instance consists of two stars of height 1 (i.e., they have two levels) where each internal node has $k-1$ children. The leaf nodes of one star are connected to the root node of the other star. An integral solution can pick at most one edge. However, a fractional solution that sets the value of every edge to $\frac{1}{k}$ is feasible and has value $2 - \frac{2}{k}$. □

2.3 Hardness and Inapproximability Results

In this section we prove hardness of tree-constrained bipartite matching even if the weights in the matching are restricted to the values zero and one. Subsequently, we show by an approximation-preserving reduction from a restricted MAX SAT version that TCBM does not admit a PTAS.

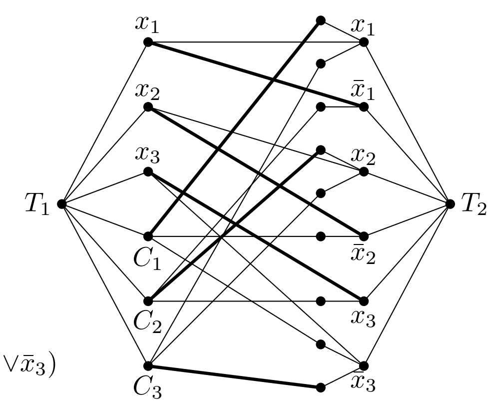
Theorem 3. *For an instance $I = (T_1, T_2, w)$ of* TCBM*, with $w : V_1 \times V_2 \mapsto \{0, 1\}$, and an integer k, it is $\mathcal{NP}$-complete to decide whether there exists a tree-constrained bipartite matching of weight at least k.*

Proof. Clearly, the problem is in $\mathcal{NP}$. To prove that it is $\mathcal{NP}$-hard, we devise a polynomial-time reduction τ from SAT, the problem of deciding whether a Boolean formula has a satisfying assignment. Given a CNF formula ϕ with m clauses over n variables, we construct a TCBM instance $I = (T_1, T_2, w)$, such that ϕ is satisfiable if and only if I admits a matching of weight $n + m$.

Tree T_1 is the star S_{n+m}, with one leaf per variable and clause. Depth-2 tree T_2 has for each literal occurring in ϕ a node at level 1. Such a node, corresponding to some literal l_k, has a child node for each occurrence of literal l_k in ϕ. What remains is the definition of the weight function w. For each leaf u in T_1, representing some variable x_i, we define $w(u, v) := 1$ and $w(u, v') := 1$, where v, v' are level-1 nodes in T_2 that correspond to literals x_i and $\neg x_i$, respectively. For each leaf u in T_1, representing some clause C_i of ϕ, we set, for all literals l_j occurring in C_i, $w(u, v) := 1$, where v is a child node of a level-1 node in T_2 that corresponds to literal l_j. Hereby we pick distinct child nodes v in T_2, such that each level-2 node is incident to exactly one edge of weight 1. All remaining weights are set to 0. See Figure 4 for an illustrative example of this construction.

$(x_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \bar{x}_3)$

(a)



(b)

Fig. 4. Illustrating the construction of an TCBM instance from a SAT instance using an example. (a) SAT instance in conjunctive normal form. (b) Transformed tree-constrained bipartite matching instance, only edges with unit weight are shown. A maximum weight tree-constrained bipartite matching, which corresponds to a satisfying truth assignment, is shown in bold.

First, we show that if ϕ is satisfiable, then $\tau(\phi)$ admits a matching $\mathcal{M}$ of weight $m + n$. For this, let ν be a satisfying assignment for ϕ. Starting from $\mathcal{M} = \emptyset$ we add, for each variable x_i, edge (u, v) to $\mathcal{M}$, where u is the leaf in T_1 representing variable x_i, and v the level-1 node in T_2 representing either literal x_i, if $\nu(x_i) =$ false, or literal $\neg x_i$, if $\nu(x_i) =$ true. Additionally we add, for each clause C_j, an edge (u', v') to $\mathcal{M}$, where u' is a leaf in T_1 representing C_j, and v' is an unmatched child node of a level-1 node $\hat{v}$ in T_2 representing a literal l_k contained in C_j that evaluates to true under the assignment ν. Note that $\hat{v}$ has not been matched before in this case and thus the matching of a child node is valid. Since in each clause at least one literal evaluates to true, the resulting matching $\mathcal{M}$ has weight $m + n$.

Since in every matching $\mathcal{M}$ of weight $m + n$ for $\tau(\phi)$ each leaf in T_1 is matched by an edge of weight 1, one can always derive a satisfying assignment for ϕ from $\mathcal{M}$. Therefore, if ϕ is not satisfiable, the weight of a maximum weight matching in $\tau(\phi)$ is at most $m + n - 1$. ☐

We next prove that TCBM is $\mathcal{APX}$-hard. The reduction is made from 3-OCC-MAX 2SAT, a restricted form of MAX SAT, where each clause contains two literals and each variable occurs at most three times.

Theorem 4 ([4]). *For any* $\epsilon > 0$ *it is* $\mathcal{NP}$*-hard to decide whether an instance of* 3-OCC-MAX 2SAT *with* $2016n'$ *clauses (and* $1344n'$ *variables) has a truth assignment that satisfies at least* $(2012-\epsilon)n'$ *clauses, or at most* $(2011+\epsilon)n'$*.*

Theorem 5. *For any* $\epsilon > 0$*, it is* $\mathcal{NP}$*-hard to approximate* TCBM *within factor* $6044/6043 - \epsilon$*.*

Proof. Our reduction τ' from 3-OCC-MAX 2SAT to TCBM differs from reduction τ described in the proof of Theorem 3 only in the definition of the weight function w. For leaves u in T_1 representing some variable x_i and level-1 nodes v in T_2 corresponding to literal x_i or $\neg x_i$, we set $w(u,v) := 3$, and leave w unchanged otherwise. Then it is easy to see that the maximum number of satisfiable clauses in ϕ is k if and only if the maximum weight of a tree-constrained matching in $\tau'(\phi)$ is $3n + k$. Since the instance constructed in [4] uses $1344n'$ variables, $k = (2012-\epsilon)n'$ and $k = (2011+\epsilon)n'$ correspond to tree constrained matchings of weight $(6044-\epsilon)n'$ and $(6043+\epsilon)n'$ respectively. □

3 Matching Posets

In this section we investigate the more general problem of matching posets:

Definition 2 (Poset-constrained k-partite matching problem, k-PCM). *Given k posets $\mathcal{P}_i = (V_i, \preceq_i)$, $1 \le i \le k$, and a weight function $w : V_1 \times V_2 \times \cdots \times V_k \mapsto \mathbb{R}_+$, find a maximum weight k-dimensional matching $\mathcal{M}$ in the complete k-partite k-uniform hypergraph $H = (V_1, \ldots, V_k, E)$ with hyper-edge weights induced by w, such that $(p_1, \ldots, p_k) \in \mathcal{M}$ implies $(q_1, \ldots, q_k) \notin \mathcal{M}$ if there exists $1 \le i \le k$ with q_i being comparable to p_i in $\mathcal{P}_i$.*

3.1 A Fractional Local Ratio Algorithm

Unlike the tree case, this problem cannot be directly reduced to MWIS in k-interval graphs, and therefore, the $2k$-approximation of Bar-Yehuda et al. [3] does not readily apply. However, we show that the fractional local ratio technique can still be used to solve the poset case. We work with the following linear programming formulation.

$$\max \sum_{p \in E} w_p x_p \tag{MP}$$

$$\text{s.t.} \sum_{p : p_i \in C} x_p \le 1 \qquad \forall \text{ chain } C \text{ in } \mathcal{P}_i, i = 1, \ldots, k \tag{6}$$

$$x_p \ge 0 \qquad \forall\, p \in E.$$

We remark that even though the above linear program is exponentially large, there is a simple separation oracle based on the polynomial time algorithm for computing a longest path in an acyclic directed graph. In Section 4 we show an alternative linear-size formulation.

As we did before in the tree case, the crux of the analysis is to show that there is an edge $p \in E$ with low fractional local ratio. Our bound will depend on the *maximum upward independence number* of the individual posets.

Definition 3. *For a given poset $\mathcal{P} = (V, \preceq)$, the maximum upward independence number of $\mathcal{P}$, denoted by $\rho(\mathcal{P})$ is defined as*

$$\underset{v \in V}{\text{maximum}} \ \textit{size of largest antichain in } (\{u \in V : v \preceq u\}, \preceq).$$

We show that the fractional local ratio of any feasible solution is bounded by the maximum upward independence number of the posets in our instance.

Lemma 6. *Let* $\mathbf{x}$ *be a feasible solution to (MP). There is some* $p \in E$ *such that*

$$\sum_{q\,:\,\exists i \bullet q_i \preceq_i p_i \vee p_i \preceq_i q_i} x_q \leq 2 \sum_i \rho(\mathcal{P}_i).$$

Notice that if we consider the poset $\mathcal{P}$ induced by some tree T, the maximum upward independence number is 1. This is because for any vertex v of T, the poset induced by $\{u \in T : v \preceq u\}$ is a total order; namely, the path from v to the root of the tree.

Proof (Lemma 6). For $p \in E$ and $i = 1, \ldots, k$ let $\mathrm{CP}(p, i)$ be a minimum size chain partition of the poset $(\{u \in V_i : p_i \preceq_i u\}, \preceq_i)$. Then we have,

$$\sum_p x_p \sum_{q\,:\,\exists i \bullet p_i \preceq_i q_i \vee q_i \preceq_i p_i} x_q = 2 \sum_p x_p \sum_{q : \exists i \bullet p_i \preceq_i q_i} x_q \leq 2 \sum_p x_p \sum_i \sum_{C \in \mathrm{CP}(p,i)} \sum_{q : q_i \in C} x_q,$$

$$\leq 2 \sum_p x_p \sum_i \sum_{C \in \mathrm{CP}(p,i)} 1 \leq \left(2 \sum_i \rho(\mathcal{P}_i) \right) \sum_p x_p,$$

where the first and second lines follow by simply re-arranging the terms in the sum. The third line follows from constraint (6) and the forth from Dilworth's Theorem, which states that the size of the largest antichain in a poset equals the size of the smallest chain partition. Since all the x_p are non-negative, the lemma follows. □

Using the fractional local ratio framework of Bar-Yehuda et al. [3] we immediately obtain the following result.

Theorem 6. *There is a* $2 \sum_i \rho(\mathcal{P}_i)$ *approximation algorithm for k-PCM.*

It can be shown that the dependency on $\sum_i \rho(\mathcal{P}_i)$ in the approximation ratio is necessary for any algorithm based on the linear program (MP).

Lemma 7. *There are instances of k-PCM where the integrality gap of* (MP) *is* $\left(1 - \frac{1}{k}\right) \sum_i \rho(\mathcal{P}_i)$ *for arbitrary large* k.

Proof. Let ℓ be a prime number. Consider a $\ell \times \ell$ grid of ℓ^2 points. For each $i = 1, \ldots, \ell$ we define a precedence constraints $\preceq_i$ as follows. For two points (x, y) and (x', y') in the grid, if $y < y'$ and $x' \equiv x + i(y' - y) \mod \ell$, then $(x, y) \preceq_i (x', y')$. Also, we define a precedence constraint $\preceq_{\ell+1}$ as follows. For two points (x, y) and (x', y') in the grid, if $y = y'$ and $x < x'$, then $(x, y) \preceq_{\ell+1} (x', y')$. The posets construction is illustrated in Figure 5.

For each point (x, y) we join the $\ell + 1$ copies of (x, y) (one copy for each poset) with a hyper edge of weight 1. All other hyper edges have weight 0. Thus, the $(\ell + 1)$-PCM problem in this case reduces to finding a maximum set of points S from the grid such that S is simultaneously independent in each poset $\mathcal{P}_i$ for $i = 1, \ldots, \ell + 1$. For this reason, any feasible integral solution can pick at most one hyper edge: For any two points (x, y) and (x', y') there exists a poset $\mathcal{P}_i$ such that $(x, y) \preceq_i (x', y')$. On the other hand, picking a $\frac{1}{\ell}$ fraction of every hyper edge with weight 1 is a feasible fractional solution: The maximum length chain in each of the posets is ℓ. Since there are ℓ^2 edges with weight 1, it follows that the integrality gap of the instance is at least ℓ. Furthermore, it is easy to see that for each poset $\rho(\mathcal{P}_i) = 1$ for $i = 1, \ldots, \ell + 1$, yielding the desired result. □

3.2 Hardness

In this section we show that the dependence of the approximation factor on the maximum poset width $\rho(\mathcal{P})$ is unavoidable, by showing that, under plausible complexity assumptions, 2-PCM is hard to approximate within $2^{\log^{1-\epsilon} \rho}$ for any $\epsilon > 0$, where ρ is the maximum width of the posets.

We will use a reduction from the *maximum label-cover* problem [1]. For convenience we use the following equivalent definition [10].

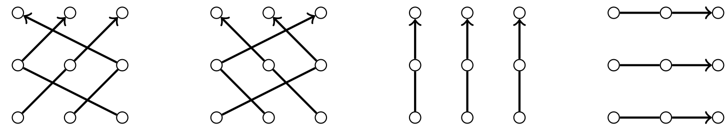


Fig. 5. Integrality gap construction from Lemma 7 for $k = 4$.

Definition 4 (MAXREP). *Given a bipartite graph $G = (A, B, E)$, with a partition of A and B into k disjoint sets $A_1, \dots, A_k$ and $B_1, \dots, B_k$, respectively, find subsets of vertices $A' \subseteq A$ and $B' \subseteq B$, such that, $|A' \cap A_i| \le 1$ and $|B' \cap B_i| \le 1$, for $i = 1, \dots, k$, so as to maximize the number of edges*

$$E(A', B') := \{\{a, b\} \in E \ : A' \cap A_i = \{a\} \text{ and } B' \cap B_j = \{b\} \text{ for some } i,\ j\}.$$

Theorem 7 ([6, 10]). MAXREP *on a graph with $|A| = |B| = n$ cannot be approximated within a factor of $2^{\log^{1-\epsilon} n}$, for any $\epsilon > 0$, unless $NP \subseteq DTIME(n^{\text{polylog}\, n})$.*

Theorem 8. *For any $\epsilon > 0$ and $\rho := \max\{\rho(\mathcal{P}_1), \rho(\mathcal{P}_2)\}$ there is no $2^{\log^{1-\epsilon} \rho}$-factor approximation for 2-*PCM *unless $NP \subseteq DTIME(n^{\text{polylog}\, n})$.*

Proof. Given any instance $G = (A = A_1 \cup \ldots \cup A_k, B = B_1 \cup \ldots \cup B_k, E)$ of MAXREP, we construct an unweighted instance $I = (\mathcal{P}_1, \mathcal{P}_2, E_I)$ of 2-PCM, such that $\rho := \max\{\rho(\mathcal{P}_1), \rho(\mathcal{P}_2)\} \le n^2 = |A|^2 = |B|^2$, and for any feasible solution on G there is a feasible solution on I with at least the same objective value, and vice versa. This would obviously imply the statement of the theorem, since a $2^{\log^{1-\epsilon} \rho}$-approximation for I, for some $\epsilon > 0$, would imply a $2^{\log^{1-\frac{\epsilon}{2}} n}$-approximation for G (for sufficiently large n).

We define two posets $\mathcal{P}_1$ and $\mathcal{P}_2$ that correspond to A and B, respectively. Both posets will have precedence graphs of the *series-parallel* type. We show the construction for the first poset $\mathcal{P}_1$. For every vertex $a \in A$, we define a height-two subposet $\mathcal{P}(a)$ consisting of d parallel chains, where $d = \deg_G(a)$ is the degree of a in G as follows. If the neighbors of a in G are $\{b_1, \dots, b_d\}$, the elements of $\mathcal{P}(a)$ are $a^\perp, a^{b_1}, \dots, a^{b_d}, a^\top$, where $a^\perp \prec_1 a^{b_i} \prec_1 a^\top$, for all $i = 1, \dots, d$.

For each $i \in [k]$, we chain together (in an arbitrary order) the posets $\mathcal{P}(a)$ corresponding to $a \in A_i$, and then we connect all these chained posets together in parallel to obtain the whole poset $\mathcal{P}_1$. More precisely, we define two more elements $A^\perp$ and $A^\top$. For each $i \in [k]$, if the elements of set A_i are $a_1, \dots, a_r$, then we define a subposet $\mathcal{Q}_i$, such that, for $i = 1, \dots, r-1$, $a_i^\top = a_{i+1}^\perp$, $A^\perp \prec_1 a_1^\perp$, and $a_r^\top \prec_1 A^\top$. This finishes the definition of poset $\mathcal{P}_1$; poset $\mathcal{P}_2$ is defined similarly. The edges between the two different posets are connected according to the edges in the original graph in the obvious way: $E_I = \{(a^b, b^a) \ : \ a \in A, b \in B\}$. An example of the construction is given in Figure 6.

Let us now show that the above construction preserves the sizes of the solutions. Given a feasible solution $(A' \subseteq A, B' \subseteq B)$ for G, we define the corresponding solution $\mathcal{M} \subseteq E_I$ in I as: $\mathcal{M} = \{(a^b, b^a) : \ \{a, b\} \in E(A', B')\}$. It follows easily from the construction, since $|A' \cap A_i| \le 1$ and $|B' \cap B_j| \le 1$ for all i, j, that $\mathcal{M}$ is a feasible solution for I, and obviously $|\mathcal{M}| = |E(A', B')|$. Conversely, given a feasible matching $\mathcal{M}$ in I, we construct a solution (A', B') for G, where $A' = \{a \in A \ : \exists (a^b, b^a) \in \mathcal{M} \text{ for some } b \in B\}$ and $B' = \{b \in B \ : \exists (a^b, b^a) \in \mathcal{M} \text{ for some } a \in A\}$. Clearly, $E(A', B') \ge |\mathcal{M}|$, and the construction of the posets and the definition of $\mathcal{M}$ guarantees that $|A' \cap A_i| \le 1$ and $|B' \cap B_j| \le 1$, for all i and j.

Finally, let us note that

$$\rho(\mathcal{P}_1) = \max_{\substack{A' \subseteq A \\ \forall i:\ |A' \cap A_i| = 1}} \sum_{a \in A'} \deg_G(a) \ \text{ and } \ \rho(\mathcal{P}_2) = \max_{\substack{B' \subseteq B \\ \forall j:\ |B' \cap B_j| = 1}} \sum_{b \in B'} \deg_G(b),$$

and hence ρ is bounded by $|E| \le n^2$, as claimed. $\square$

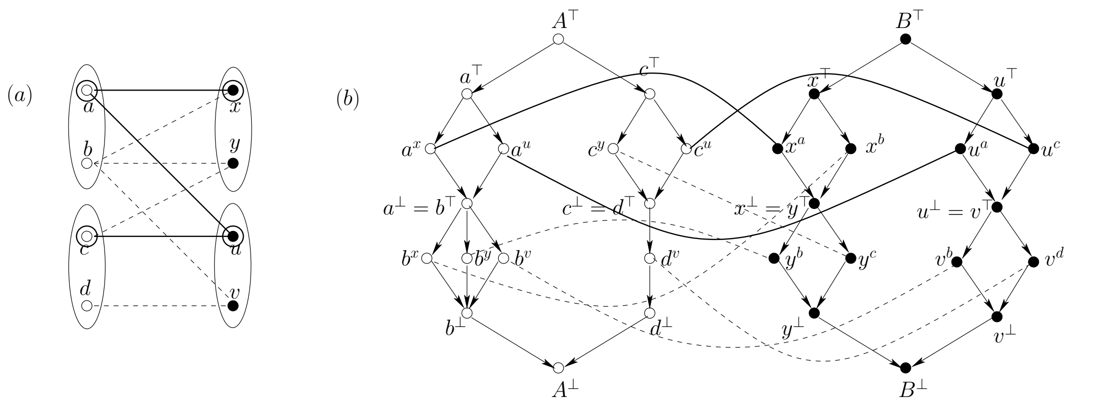


Fig. 6. Illustrating the construction of an 2-PCM instance from a MAXREP instance using an example. (a) a 2-PCM instance: $A_1 = \{a, b\}$, $A_2 = \{c, d\}$,$B_1 = \{x, y\}$, and $B_2 = \{u, v\}$; $A' = \{a, c\}$ and $B' = \{x, u\}$ gives a solution with value 3. (b) The corresponding 2-PCM instance, only edges with unit weight are shown. The solution corresponding to A', B' is is shown in bold.

4 Alternative LP formulation for Matching Posets

We now turn our attention to solving the linear program. As already mentioned, MP can be solved using the Ellipsoid algorithm and a separation oracle based on longest paths for directed acyclic graphs. Since the Ellipsoid method is very slow in practice, we provide an alternative more compact LP formulation. First let us recast the previous formulation

$$\max \sum_{p \in E} w_p x_p \tag{MP'}$$

$$\text{s.t.} \sum_{p : p_i = u} x_p = y_u \qquad \forall u \in V_i, i = 1, \dots, k \tag{7}$$

$$\sum_{u \in C} y_u \leq 1 \qquad \forall \text{ chain } C \text{ in } \mathcal{P}_i, i = 1, \dots, k \tag{8}$$

$$x_p, y_u \geq 0 \qquad \forall\, p \in E, u \in V_1 \cup \dots \cup V_k.$$

The idea is to express constraints (8) for a fixed i, with a single constraint over the outcome of an optimization problem. For the purpose of simplifying our formulation, we augment each poset $\mathcal{P}_i$ by setting

$$V_i' = V_i \cup \{s_i, t_i\} \text{ and } \preceq_i' = \preceq_i \cup \{(s_i, u) : u \in V_i\} \cup \{(u, t_i) : u \in V_i\}.$$

Now consider the following flow-based formulation for finding a maximum length s_i-t_i path in $\mathcal{P}_i$ for a fix set of y_u values (we assume $y_{s_i} = y_{t_i} = 0$).

$$\max \sum_{u \in V_i'} y_u \left(\sum_{v : v \preceq_i' u} z_{v,u} \right) \tag{HP}$$

$$\text{s.t.} \sum_{v : v \preceq_i' u} z_{v,u} - \sum_{v : u \preceq_i' v} z_{u,v} = 0 \qquad u \in V_i \tag{9}$$

$$\sum_{u \in V_i} z_{u,t_i} = 1 \tag{10}$$

$$\sum_{u \in V_i} -z_{s_i,u} = -1 \tag{11}$$

$$z_{u,v} \geq 0 \qquad \forall\, u, v \in V_i' : u \preceq_i' v$$

Since the graph induced by the poset $\mathcal{P}'_i$ is acyclic, the above linear program is guaranteed to be integral. Therefore, if we were to replace all Constraint (8) involving $\mathcal{P}_i$ with a single constraint asking that the value of the above program be no more than 1, we would have an alternative formulation. This change, however, does not give us a linear program. To that end, consider the dual of (HP).

$$
\begin{aligned}
\min\ & p_{t_i} - p_{s_i} && && \text{(DHP)}\\
\text{s.t. } & p_v - p_u \geq y_v && \forall\, u, v \in V'_i : u \preceq'_i v && (12)\\
& p_u \text{ free} && \forall\, u \in V'_i && (13)
\end{aligned}
$$

Since the value of any feasible dual solution is an upper bound on the optimal value of the primal problem, we can express our constraint on the length of the longest path.

$$
\begin{aligned}
\max\ & \sum_{p \in E} w_p x_p && && \text{(MP")}\\
\text{s.t. } & \sum_{p : p_i = u} x_p = y_u && \forall u \in V_i, i = 1, \ldots, k && (14)\\
& y_{s_i} = y_{t_i} = 0 && \forall\, i = 1, \ldots, k && (15)\\
& p_{t_i} - p_{s_i} \leq 1 && \forall\, i = 1, \ldots, k && (16)\\
& p_v - p_u \geq y_v && \forall\, u, v \in V'_i : u \preceq'_i v,\ i = 1, \ldots, k && (17)\\
& x_p \geq 0 && \forall\, p \in E && (18)\\
& y_u \geq 0 && \forall u \in V'_1 \cup \cdots \cup V'_k &&\\
& p_u \text{ free} && \forall u \in V'_1 \cup \cdots \cup V'_k &&
\end{aligned}
$$

Notice that the projection of the feasible region of MP" to the (x, y) variables is precisely the feasible region of (MP'). Thus, we arrive at a pure linear programming formulation for the k-PCM problem whose size is linear on the size of the posets and the hypergraph connecting the different posets.

5 Concluding remarks

Perhaps the most general version of problems TCBM and k-PCM is the following:

Definition 5 (Graph-constrained k-partite matching problem, k−GCM). *Given k graphs $G_i = (V_i, E_i)$, $1 \leq i \leq k$, and a weight function $w : V_1 \times V_2 \times \cdots \times V_k \mapsto \mathbb{R}_+$, find a maximum weight k-dimensional matching $\mathcal{M}$ in the complete k-partite k-uniform hypergraph $H = (V_1, \ldots, V_k, E)$ with hyper-edge weights induced by w, such that $(p_1, \ldots, p_k) \in \mathcal{M}$ implies $(q_1, \ldots, q_k) \notin \mathcal{M}$ if there exists $1 \leq i \leq k$ with either $\{p_i, q_i\} \in E_i$ or $p_i = q_i$.*

Note that k−GCM specializes to k−PCM when all the G'_is belong to the class of *comparability* graphs, which is a sublcass of *perfect* graphs. For a graph $G = (V, E)$, denote by $\alpha(G)$ the independence number of G, and for $v \in V$, denote by $G(v)$ the subgraph of G induced on the (open) neigbourhood $\{u \in V : \{v, u\} \in E\}$ of v. Theorem 6 admits the following generalization.

Theorem 9. *If all the G'_is are perfect, then there is a $\sum_i \rho(G_i)$ approximation algorithm for k-GCM, where $\rho(G_i) := \text{maximum}_{v \in V_i}\, \alpha(G_i(v))$.*

We work with the linear programming formulation with clique constraints.

$$\max \sum_{p\in E} w_p x_p \tag{P1}$$

$$\text{s.t.} \sum_{p:p_i\in C} x_p \le 1 \qquad \forall \text{ clique } C \text{ in } G_i, i=1,\dots,k \tag{19}$$

$$x_p \ge 0 \qquad \forall\, p \in E.$$

Again, there is a separation oracle based on the polynomial time algorithm for computing a maximum weight clique in perfect graphs [7].

To prove Theorem 9, it is enough to prove the following generalization of Lemma 6.

Lemma 8. *Let* $\mathbf{x}$ *be a feasible solution to (P1). There exists some* $p \in E$ *such that*

$$\sum_{q\,:\,\exists i\,\bullet\,\{q_i,p_i\}\in E_i \vee p_i=q_i} x_q \le \sum_i \rho(G_i).$$

Proof. Denote by $G_i[v]$ the subgraph of G_i induced on the closed neighborhood $\{v\} \cup \{u \in V_i : \{v,u\} \in E_i\}$ of v. For $p \in E$ and $i = 1,\dots,k$, note that $G_i[p_i]$ is also perfect and therefore it has a clique partition $\mathrm{CP}(p,i)$ of size $\alpha(G_i[p_i])$. Then we have,

$$\begin{aligned}\sum_p x_p \sum_{q\,:\,\exists i\,\bullet\,\{p_i,q_i\}\in E_i \vee p_i=q_i} x_q &\le \sum_p x_p \sum_i \sum_{C\in \mathrm{CP}(p,i)} \sum_{q:q_i\in C} x_q,\\ &\le \sum_p x_p \sum_i \sum_{C\in\mathrm{CP}(p,i)} 1,\\ &\le \left(\sum_i \rho(G_i)\right)\sum_p x_p.\end{aligned}$$

The lemma follows. □

A graph G is said (see, e.g., [9]) to have an *inductive independence number* $\rho := \rho(G)$ if there exists an ordering on the vertices s.t. for each vertex v, the subgraph induced on the neighbors of v (let us say, for simplicity, including v itself), that precede v in the order, has independence number at most ρ. For instance, the intersection graph of a set of fat objects in the plane (e.g. disks or squares) has a small inductive independence number.

Theorem 10. *There is a* $2\sum_i \rho(G_i)$ *approximation algorithm for the* k-GCM, *where* $\rho(G_i)$ *is the inductive independence number of graph* G_i.

Let π_i be the ordering on the vertices given for graph G_i. Following [9], we use the following linear programming formulation.

$$\max \sum_{p\in E} w_p x_p \tag{P2}$$

$$\text{s.t.} \sum_{\substack{p:\{p_i,q_i\}\in E_i \vee p_i=q_i\\ \pi_i(p_i)\le\pi_i(q_i)}} x_p \le \rho(G_i) \qquad \forall\, q\in E, i=1,\dots,k \tag{20}$$

$$0 \le x_p \le 1 \qquad \forall\, p \in E.$$

Again, it is enough to prove the following lemma.

Lemma 9. *Let* $\mathbf{x}$ *be a feasible solution to (P2). There exists some* $p \in E$ *such that*

$$\sum_{q\,:\,\exists i\,\bullet\,\{q_i,p_i\}\in E_i \vee p_i=q_i} x_q \le 2\sum_i \rho(G_i).$$

Proof. By (20) we have,

$$\sum_p x_p \sum_{q\,:\,\exists i \bullet \{p_i,q_i\}\in E_i \vee p_i=q_i} x_q \leq 2\sum_p x_p \sum_{q\,:\,\exists i \bullet \{p_i,q_i\}\in E_i,\ \pi_i(p_i)\leq\pi_i(q_i)} x_q$$
$$\leq \left(2\sum_i \rho(G_i)\right)\sum_p x_p,$$

and the lemma follows. □

Acknowledgments. We thank Axel Mosig for introducing us to the problem and for helpful discussions. We also thank an anonymous reviewer of an earlier version for pointing out the connection between our problem and the work of Bar-Yehuda et al. [3]. Thanks also to Yuk Hei Chan for helpful discussions.

References

[1] S. Arora, L. Babai, J. Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *Journal of Computer and System Sciences*, 54(2):317 – 331, 1997.

[2] A. Bar-Noy, R. Bar-Yehuda, A. Freund, J. Naor, and B. Schieber. A unified approach to approximating resource allocation and scheduling. *Journal of the ACM*, 48:1069–1090, 2001.

[3] R. Bar-Yehuda, M. M. Halldórsson, J. Naor, H. Shachnai, and I. Shapira. Scheduling split intervals. *SIAM Journal on Computing*, 36(1):1–15, 2006.

[4] P. Berman and M. Karpinski. On some tighter inapproximability results. In *Proc. of ICALP*, pages 200–209, 1999.

[5] S. Canzar, K. Elbassioni, G. W. Klau, and J. Mestre. On tree-constrained matchings and generalizations. In *Proc. 38th International Colloquium on Automata, Languages and Programming (ICALP 2011)*, Lecture Notes in Computer Science. Springer-Verlag, 2011. To appear.

[6] U. Feige and L. Lovász. Two-prover one-round proof systems: their power and their problems. In *Proc. of STOC*, pages 733–744. ACM, 1992.

[7] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer, New York, 1988.

[8] E. Hazan, S. Safra, and O. Schwartz. On the complexity of approximating k-dimensional matching. In *Proc. of RANDOM*, pages 83–97, 2003.

[9] M. Hoefer, T. Kesselheim, and B. Vöcking. Approximation algorithms for secondary spectrum auctions. *CoRR*, abs/1007.5032, 2010.

[10] G. Kortsarz. On the hardness of approximating spanners. *Algorithmica*, 30(3):432–450, 2001.

[11] A. Mosig, S. Jäger, C. Wang, S. Nath, I. Ersoy, K. Palaniappan, and S.-S. Chen. Tracking cells in life cell imaging videos using topological alignments. *Algorithms for Molecular Biology*, 4: 10, 2009.

[12] L. A. Wolsey and G. L. Nemhauser. *Integer and Combinatorial Optimization.* Wiley-Interscience, 1999.

[13] H. Xiao, Y. Li, J. Du, and A. Mosig. Ct3d: Tracking Microglia Motility in 3D Using a Novel Cosegmentation Approach. *Bioinformatics*, 27(4):564–571, 2011.

[14] F. Yang, M. A. Mackey, F. Ianzini, G. Gallardo, and M. Sonka. Cell segmentation, tracking, and mitosis detection using temporal context. In *Proc. of MICCAI*, pages 302–309, 2005.

[15] C. Zimmer, B. Zhang, A. Dufour, A. Thebaud, S. Berlemont, V. Meas-Yedid, and J.-C. O. Marin. On the digital trail of mobile cells. *Signal Processing Magazine*, 23(3):54–62, 2006.