Graph Products Revisited: Tight Approximation Hardness of Induced Matching, Poset Dimension and More

Parinya Chalermsook* Bundit Laekhanukit† Danupon Nanongkai‡

Abstract

Graph product is a fundamental tool with rich applications in both graph theory and theoretical computer science. It is usually studied in the form $f(G*H)$ where G and H are graphs, $*$ is a graph product and f is a graph property. For example, if f is the *independence number* and $*$ is the *disjunctive product*, then the product is known to be *multiplicative*: $f(G*H) = f(G)f(H)$.

In this paper, we study graph products in the following non-standard form: $f((G \oplus H) * J)$ where G, H and J are graphs, $\oplus$ and $*$ are two different graph products and f is a graph property. We show that if f is the *induced and semi-induced matching number*, then for some products $\oplus$ and $*$, it is *subadditive* in the sense that $f((G \oplus H) * J) \leq f(G * J) + f(H * J)$. Moreover, when f is the *poset dimension number*, it is *almost* subadditive.

As applications of this result (we only need $J = K_2$ here), we obtain tight hardness of approximation for various problems in discrete mathematics and computer science: bipartite induced and semi-induced matching (a.k.a. maximum expanding sequences), poset dimension, maximum feasible subsystem with 0/1 coefficients, unit-demand min-buying and single-minded pricing, donation center location, boxicity, cubicity, threshold dimension and independent packing.

1 Introduction

Graph products generally refer to a way to use two graphs, say G and H, to construct a new graph, say $G * H$ for some product $*$. Studying properties of graphs resulting from a graph product, i.e., $f(G * H)$ for some function f, has been an active research area with countless applications in graph theory and computer science. For example, the fact that the *independence number* α of the *disjunctive product* $G \vee H$ is *multiplicative*, i.e., $\alpha(G \vee H) = \alpha(G)\alpha(H)$, has been used to amplify the hardness of approximating the maximum independent set problem.

In this paper, we study some graph properties when we apply graph products in a non-standard fashion to improve approximation hardness of several problems. We will study graph products in the form $(G \oplus H) * J$ where G, H and J are any graphs and $*$ and $\oplus$ denote two different products. This form may look strange at first, but it will be clear later that it in fact captures a "generic" graph transformation technique that has been used a lot in the past (cf. Section 3).

*IDSIA, Lugano, Switzerland. Supported by the Swiss National Science Foundation project 200020-122110/1 and by Hasler Foundation Grant 11099. Partially supported by Julia Chuzhoy's CAREER grant CCF-0844872. E-mail: parinya@cs.uchicago.edu

†School of Computer Science, McGill University, Montreal QC, Canada. Supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) grant No. 28833 and by European Research Council (ERC) Starting Grant 279352. E-mail: blaekh@cs.mcgill.ca

‡Theory and Applications of Algorithms Research Group, University of Vienna, Austria, and Nanyang Technological University, Singapore. E-mail: danupon@gmail.com.

The products we will study are the *tensor product* $G \times H$, the *extended tensor product*[1] $G \times_e H$, the *disjunctive product* $G \vee H$ and the *lexicographic product* $G \cdot H$. These products produce a graph whose vertex set is $V(G) \times V(H) = \{(u, v) : u \in V(G), v \in V(H)\}$ with different edge sets. Their exact definitions are not necessary at this point, but if you are impatient, see Section 2.

Properties of a graph G that we are interested in are the *induced matching number* $\mathsf{im}(G)$, the *semi-induced matching number* $\mathsf{sim}(G)$ and the *poset dimension number* $\mathsf{dim}(G)$. Informally, an induced matching of an undirected graph G is a matching $\mathcal{M}$ of G such that no two edges in $\mathcal{M}$ are joined by an edge in G. We let the induced matching number of G, denoted by $\mathsf{im}(G)$, be the size of the maximum induced matching. See the formal definition in Section 2. Definitions of *poset* and other graph properties are not needed in this section and are deferred to Section 2.

Now that we have introduced all notations necessary, we are ready to state our result. We show that for $f = \mathsf{im}$ or $f = \mathsf{sim}$ and for an appropriate choice of product $\oplus$ and $*$, f will be *subadditive*, i.e., $f((G \oplus H) * J) \leq f(G * J) + f(H * J)$ for any graphs G, H and J. For $f = \mathsf{dim}$, we will get *almost* subadditivity instead. The precise statement is as follows.

Theorem 1.1 ((Almost) Subadditivity)**.** *For any undirected graphs G, H and J and a height-two poset $\vec{P}$,*

$$\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(G \times J) + \mathsf{im}(H \times J) \tag{1}$$
$$\mathsf{sim}((G \vee H) \times J) \leq \mathsf{sim}(G \times J) + \mathsf{sim}(H \times J) \tag{2}$$
$$\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}(G \times_e \vec{P}) + \chi(G)\mathsf{dim}(H \times_e \vec{P}) + \mathsf{dim}(\vec{P}) \tag{3}$$

where $\chi(G)$ is the chromatic number of G.

Note that Eq.(3) suggests that the poset dimension number is almost subadditive in the sense that if $\chi(G)$ and $\mathsf{dim}(\vec{P})$ are small, then it will be subadditive (with a small multiplicative factor). This will be the case when we use it to prove the hardness of approximation; see Section 5 for more detail.

Organization. In Subsection 1.1, we give an example of how Theorem 1.1 plays a role in proving hardness of approximation. In Subsection 1.2, we discuss problems whose hardness of approximation can be obtained via our technique. In Section 2, we define formal terms needed in the rest of the paper. We then prove Theorem 1.1 in Section 3 (for the special case which gives more intuition) and Section 4 (for the general case). Section 5 and 6 show the hardness results. Our results are summarized in Fig. 1.

1.1 Using Subadditivity (Theorem 1.1).

We now sketch the proof idea of the $n^{1-\epsilon}$ hardness of the bipartite induced matching problem, where n is the number of vertices, which shows how Theorem 1.1 plays a role in proving the hardness of approximation. The full proof appears in Section 5. We build on the idea of [17] and apply Theorem 1.1 with $J = K_2$.

For any graph G, let $\alpha(G)$ be the size of the maximum independent set. We use the following connection between independence and induced matching numbers which was implicitly shown in [17].

$$\alpha(G) \leq \mathsf{im}(G \times_e K_2) \leq \mathsf{im}(G \times K_2) + \alpha(G). \tag{4}$$

[1]To the best of our knowledge, this product has not been considered before. Interestingly, it is mentioned in [27, pp 42] as "not worthy of attention".

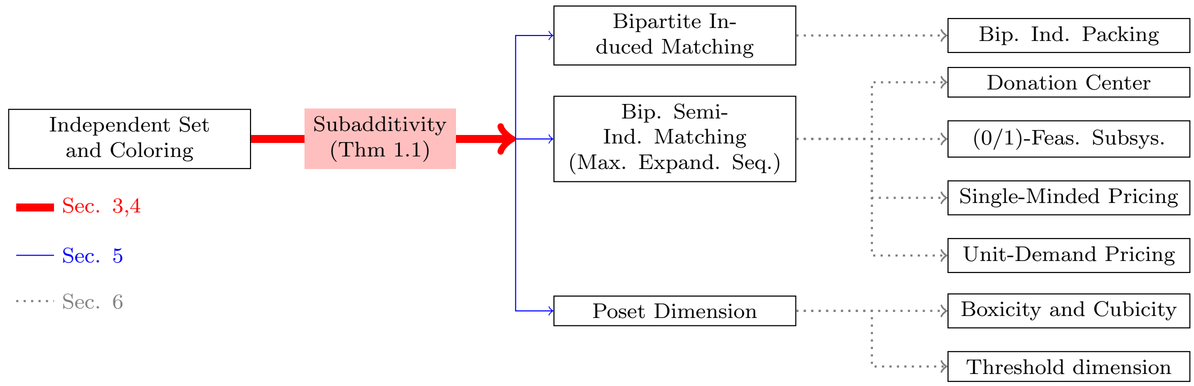


Figure 1: Paper Outline. Problems in the third column are defined in Section 2. Other problems are defined in Section 6.

If $\mathsf{im}(G \times K_2)$ is relatively small, i.e., $\mathsf{im}(G \times K_2) = O(\alpha(G))$, then we will already have the hardness of $n^{1-\epsilon}$ using the hardness of approximating the independent set number (e.g., [28]). However, $\mathsf{im}(G \times K_2)$ could be as large as $|V(G)|$, and in such case, we do not get any hardness result (not even NP-hardness). To remedy this, we apply Eq.(1) in Theorem 1.1 repeatedly to show that

$$\mathsf{im}((G^k) \times K_2) \leq \mathsf{im}((G^{k-1}) \times K_2) + \mathsf{im}(G \times K_2) \leq \ldots \leq \mathsf{im}(G \times K_2)k$$

where $G^k = G \vee G \vee \ldots \vee G$ is a k-fold product. Combining this with Eq.(4), we have

$$\alpha(G^k) \leq \mathsf{im}((G^k) \times_e K_2) \leq \mathsf{im}(G \times K_2)k + \alpha(G^k)\,.$$

It is well known that $\alpha(G^k) = (\alpha(G))^k$. Thus, after applying the k-fold product, the term $\mathsf{im}(G \times K_2)$ only grows linearly in terms of k, while the term $\alpha(G^k)$ grows exponentially! So, by choosing large enough k, the induced matching number and the independence number coincide, i.e., $\mathsf{im}((G^k) \times_e K_2) \approx \alpha(G^k)$. Now, any hardness of approximating the independence number implies immediately roughly the same hardness of approximating the induced matching number. Note that it can be checked with the definition of $\times_e$ in Section 2 that $(G^k) \times_e K_2$ is a bipartite graph, so we get the hardness of the bipartite induced matching problem as desired.

1.2 List of Applications.

The almost subadditivity properties shown in Theorem 1.1 are useful in proving many other hardness of approximation results as listed in the following theorem.

Theorem 1.2. *For any $\epsilon > 0$, unless $\mathsf{ZPP} = \mathsf{NP}$ there is no $n^{1-\epsilon}$-approximation algorithm, where n is the number of vertices in the input graph, for the following problems: bipartite induced and semi-induced matching (a.k.a. maximum expanding sequence), poset dimension, bipartite independent packing, donation center location, maximum feasible subsystem with 0/1 coefficients, boxicity, cubicity and threshold dimension.*

Additionally, there is no $d^{1/2-\epsilon}$-approximation algorithm for the induced and semi-induced matching problem on d-regular bipartite graphs.

Moreover, unless $\mathsf{NP} \subseteq \mathsf{ZPTIME}(n^{\text{poly}\log n})$, there is no $\log^{1-\epsilon} m$-approximation algorithm and no $k^{1/2-\epsilon}$-approximation algorithm for the single-minded and unit-demand pricing problems, where m is the number of consumers and k is the maximum consumer's set size.

Fig. 1 summarizes the results and reductions. Almost all reductions are done in a systematic way. We take a hard instance of the maximum independent set problem or the graph coloring problem. Then we perform an appropriate graph product and output a result. Depending on the applications, we need hard instances in various forms. All the results here are essentially *tight* except the hardness of $k^{1/2-\epsilon}$ and $d^{1/2-\epsilon}$ of the pricing problems and the induced-matching problem on d-regular graphs, respectively.

Remark on Stronger Results. We note that for problems having $n^{1-\epsilon}$-hardness stated in Theorem 1.2, we can actually prove a slightly stronger result: for any $\gamma > 0$, unless $\mathsf{NP} \subseteq \mathsf{ZPTIME}(n^{\text{poly}\log n})$, there is no $\frac{n}{2^{(\log n)^{3/4+\gamma}}}$-approximation algorithm. This is achieved by applying the result of Khot and Ponnuswami [33]. For the sake of presentation, we will prove only $n^{1-\epsilon}$-hardness using the result of Håstad [28].

Now, we are ready to discuss our hardness results. We provide the formal definitions of the first three problems in Section 2 and provide the definitions of the remaining problems in Section 6.

Bipartite Induced Matching. One immediate application is the *tight* $n^{1-\epsilon}$ hardness of *approximating the induced matching problem on bipartite graphs*, improving upon the previous best hardness of $n^{1/3-\epsilon}$ [17]. Our result also implies the tight hardness of the independent packing of graphs [12] as well. A similar technique can also be used to show that the induced matching problem on *d-regular bipartite graphs* is hard to approximate to within a factor of $d^{1/2-\epsilon}$, improving upon the APX-hardness of [16, 50]. (This result is not tight as the best known upper bound is $\Theta(d)$ [24].)

The notion of induced matching has naturally arisen in discrete mathematics and computer science. It is, for example, studied as the "risk-free" marriage problem in [45] and is a subtask of finding a *strong edge coloring.* This problem and its variations also have connections to various problems such as *storylines extraction* [35] and *network scheduling, gathering and testing* (e.g. [18, 45, 32, 38, 7]). The problem was shown to be NP-complete in [45, 11] and was later shown to be hard to approximate to within a factor of $n^{1/3-\epsilon}$ unless $\mathsf{NP} = \mathsf{ZPP}$ by [17]. We have sketched the proof of the tight hardness of $n^{1-\epsilon}$ in Section 1.1, and more detail can be found in Section 5.

Bipartite Semi-induced Matching (a.k.a. Maximum Expanding Sequence). The same technique used in proving the hardness of the bipartite induced matching problem can be extended (with some additional work) to its interesting variation which captures a few other problems. This variation was introduced independently by Briest and Krysta [10] as the *maximum expanding sequence problem* and by Elbassioni et al. [17] as the *bipartite semi-induced matching problem.* There it was used as an intermediate problem that captures the hardness of some important algorithmic pricing problems and the maximum feasible subsystem problem, which we will see shortly.

Poset Dimension. Another immediate application of Theorem 1.1 is the tight $n^{1-\epsilon}$ hardness of approximating the poset dimension, improving upon the hardness of $n^{1/2-\epsilon}$ of Hegde and Jain [29].

The notion of poset dimension has long been a central subject of study in discrete mathematics (e.g., [48]) and has connections with many other notions, e.g., transitive-closure spanners [39] as well as the boxicity and the threshold dimension of graphs [1]. A variant called the *fractional dimension* is shown to have a connection to some classical scheduling problems (e.g., [4]). We note that our technique also implies the tight hardness of approximating the fractional dimension of posets.

The computational complexity of the poset dimension problem was one of the twelve outstanding open problems in Garey and Johnsons treatise on NP-completeness [23]. It was independently shown by Yannakakis [49] and Lawler and Vornberger [36] that the problem is NP-complete. More recently, Hegde and Jain showed that the problem is hard to approximate to within a factor of $n^{1/2-\epsilon}$ unless $\mathsf{NP} = \mathsf{ZPP}$. Here we resolve the approximability of this problem using graph products.

We note that our result actually implies the tight hardness of approximating the dimension of *adjacency poset*. This is the notion, along with the *incidence poset*, of the dimension of posets arising from graphs. They have been extensively studied due to their connections with graph's planarity and chromatic number (e.g., [20, 43, 44]).

Unit-demand Min-buying (Udp-Min) and Single-minded (Smp) Pricing. A result that is not so immediate from Theorem 1.1 is the hardness of approximating the two combinatorial pricing problems, called Udp-Min and Smp. The tight hardness of these two problems were recently proved by Chalermsook et al. [13]. Here we give alternate proofs of the results in [13] by employing the tight hardness of an intermediate problem – the maximum expanding sequence problem, thus confirming the role of expanding sequences in the hardness of pricing problems suggested in [10].

Both Udp-Min and Smp are among the most basic pricing problems in the literature and have received a lot of attention (e.g., [10, 25, 40, 41, 6, 13]). Briest and Krysta showed the hardness of $\log^{\epsilon} m$, assuming the (rather non-standard) hardness of the bounded-degree bipartite independent set problem. To prove the hardness of Udp-Min and Smp, they introduced the maximum expanding sequence problem and showed that it can be reduced to Udp-Min and Smp. Thus, by proving the hardness of the maximum expanding sequence problem, they obtain the hardness results for these pricing problems. As mentioned in [10], this "indicates that expanding sequences are a common source of hardness for quite different combinatorial pricing problems". Chalermsook et al. [13] recently showed the tight hardness of $\log^{1-\epsilon} m$ of these problems, assuming a standard assumption (i.e., $\mathsf{NP} \not\subseteq \mathsf{DTIME}(n^{\mathrm{poly}\log n})$), by avoiding the maximum expanding sequence problem and proving the hardness of Udp-Min and Smp directly. In this paper, we revisited Briest and Krysta's original proposal to prove the hardness of these problems via the maximum expanding sequence problem. We show the hardness of approximation result for a special case[2] of the maximum expanding sequence problem via graph products, which then implies the hardness of Udp-Min and Smp. Our results confirm that the maximum expanding sequence problem is indeed the main source of hardness for both Udp-Min and Smp.

Maximum Feasible Subsystem with 0/1 Coefficients. In the maximum feasible subsystem (Mrfs) problem, we are given a system of m linear inequalities $\ell_i \leq a_i^T x \leq \mu_i$, where $a_i \in \{0,1\}^n$, and $\ell_i, \mu_i \in \mathbb{R}_+$. The goal is to find a non-negative solution $x \in \mathbb{R}^n_+$ that maximizes the number of constraints satisfied. When coefficients are not necessarily 0/1, the $m^{1-\epsilon}$-hardness of Mrfs was proved by Guruswami and Raghavendra [26][3]. Elbassioni et al. [17] showed that even in the 0/1-coefficient case, the problem has the hardness of $m^{1/3-\epsilon}$. They actually showed a gap-preserving reduction from the semi-induced matching problem to 0/1-Mrfs. This means that our hardness of the semi-induced matching problem immediately implies the tight hardness of $m^{1-\epsilon}$ for any ϵ for Mrfs. This hardness result holds even when we allow a violation of the upper bounds by at most an $O(n)$ factor. We also show the tight hardness of $\log^{1-\epsilon}(\max_{i\in[n]} \ell_i)$, matching an upper bound in [17].

Boxicity, Cubicity and Threshold Dimension of Graphs. The notion of *boxicity* arose from the study of *intersection graphs*. It was introduced by Roberts [21] and studied extensively in discrete mathematics. It also has connections to important graph theoretic measures such as treewidths [15], genuses [20, 2], crossing numbers [3] and the maximum degree of graphs [14]. In

[2]Our special case is different from that in [10]. (If we use their special case, we only obtain the hardness of $\log^{1/2-\epsilon} m$.) In particular, our special case requires that the input must be in some specific form of a *graph product.*

[3] Indeed, Guruswami and Raghavendra proved the hardness of the *Max 3LIN* problem, which can be seen as a special case of Mrfs

computer science, optimization problems on graphs with bounded boxicity (e.g., graphs arising from intervals, rectangles, and cubes) have also received a lot of attention. Adiga et al. [1] showed that the hardness of approximating poset dimension implies the hardness of approximating boxicity (and other closely related measures called *cubicity* and *threshold dimension*). Combining this with our hardness of approximating poset dimension, we get the tight $n^{1-\epsilon}$-hardness for all these problems.

Donation Center Location. In this problem, we are given a set of agents and a set of centers, where agents have preferences over centers and centers have capacities. The goal is to open a subset of centers and to assign a maximum-sized subset of agents to their most-preferred opened centers, while respecting the capacity constraints.

Huang and Svitkina [30] introduced this problem and showed an $n^{1/2-\epsilon}$ approximation hardness by a reduction from the maximum independent set problem. We show a straightforward reduction from the semi-induced matching problem, hence giving the tight $n^{1-\epsilon}$-hardness. This hardness result holds even when all agents have the same preference over centers, and each center has unit capacity.

2 Preliminaries

In this section, we define graph products and graph properties we will use. For any directed or undirected graph G, we use $V(G)$ and $E(G)$ to denote its vertex and edge sets, respectively. Note that if G is directed, then it is possible that, for some $u, v \in V(G)$, $uv \in E(G)$ but $vu \notin E(G)$. This is not the case when G is undirected. When a graph is directed, we shall put an arrow above $\vec{G}$ to emphasize that $\vec{G}$ is a directed graph.

Definition 2.1 (Graph Products)**.** *A* graph product *is a binary operation that constructs from two graphs G and H a graph with vertex set $V(G) \times V(H) = \{(u, v) : u \in V(G), v \in V(H)\}$, and the edge set is determined by adjacency of vertices of G and H.*

The graph products we study include the *tensor product* $G \times H$, the *extended tensor product* $G \times_e H$, the *disjunctive product* $G \vee H$ and the *lexicographic product* $G \cdot H$. The edge sets of these products are as belows.

$$\begin{aligned}
&\text{(tensor)} && E(G \times H) = \{(u,a)(v,b) : uv \in E(G) \ \underline{\text{and}}\ ab \in E(H)\} \\
&\text{(extended tensor)} && E(G \times_e H) = \{(u,a)(v,b) : (uv \in E(G) \ \underline{\text{or}}\ u = v) \ \underline{\text{and}}\ ab \in E(H)\} \\
&\text{(disjunctive)} && E(G \vee H) = \{(u,a)(v,b) : uv \in E(G) \ \underline{\text{or}}\ ab \in E(H)\} \\
&\text{(lexicographic)} && E(G \cdot H) = \{(u,a)(v,b) : uv \in E(G) \ \underline{\text{or}}\ (u = v \ \underline{\text{and}}\ ab \in E(H))\}
\end{aligned}$$

Definition 2.2 (Induced Matching Number, $\mathsf{im}(G)$)**.** *Let $G = (V, E)$ be any undirected graph. The* induced matching *of G is the set of edges $\mathcal{M} \subseteq E(G)$ such that $\mathcal{M}$ is a matching and no two edges in $\mathcal{M}$ are joined by an edge in G, i.e., for any edges $uu', vv' \in \mathcal{M}$, G has none of the edges in $\{uv, uv', u'v, u'v'\}$. The* induced matching number *of G, denoted by $\mathsf{im}(G)$, is the cardinality of the maximum-cardinality induced matching of G.*

Now, we shall define a variant of an induced matching called a semi-induced matching; this notion is with respect to a total order. For any finite set S, a *total order σ on S* is a bijection $\sigma : S \to [|S|]$. The total order σ gives an ordering of elements S as we may order elements $x_i \in S$ so that $\sigma(x_1) < \sigma(x_2) < \ldots < \sigma(x_n)$, where x_i is such that $\sigma(x_i) = i$, for all i.

Definition 2.3 (Semi-induced Matching, $\mathsf{sim}(G)$)**.** *Given any graph $G = (V, E)$ and any total order σ, we say that a matching $\mathcal{M}$ is a σ-semi-induced matching if, for any pair of edges $uu', vv' \in \mathcal{M}$ such that $\sigma(u) < \sigma(u')$ and $\sigma(u) < \sigma(v) < \sigma(v')$, there are no edges uv' and uv in E.*

We can check if a matching $\mathcal{M}$ is a σ-semi-induced matching as follows. First, we order edges in $\mathcal{M}$ as $u_1v_1, u_2v_2, \ldots, u_qv_q$ where, for any i, $\sigma(u_i) < \sigma(v_i)$ and $\sigma(u_1) < \sigma(u_2) < \ldots < \sigma(u_q)$. Now, $\mathcal{M}$ is a σ-semi-induced matching if and only if, for any $i < j$, $\mathcal{M}$ has no edge in $\{u_iu_j, u_iv_j\}$.

For any graph G, we define $\mathsf{sim}_\sigma(G)$ to be the size of a maximum σ-induced matching, and we define $\mathsf{sim}(G) = \max_\sigma \mathsf{sim}_\sigma(G)$. In the *semi-induced matching problem*, we are given a graph G, and the goal is to compute $\mathsf{sim}(G)$.

Definition 2.4 (Partially Ordered Set (poset))**.** *A directed graph $\vec{P}$ is a* partially ordered set (poset)[4] *if it is directed, acyclic and transitive (i.e., $uv, vw \in E(\vec{P}) \Rightarrow uw \in E(\vec{P})$).*

An important class of posets is a height-two poset. Given a poset $\vec{P}$, we say that a vertex is *minimal* (resp., *maximal*) if it has zero in-degree (resp., out-degree) in $\vec{P}$. A poset is a *height-two* poset if every vertex is minimal or maximal (some vertex might be both).

A poset can be defined by an ordering of d-dimensional points. This leads to the notion of the poset dimension number of graphs. For any d-dimensional points $p, q \in \mathbb{R}^d$, we say that $p < q$ if for any $1 \leq i \leq d$, $p[i] \leq q[i]$, and there exists j such that $p[j] < q[j]$; otherwise, we say that $p \not< q$.

Definition 2.5 (Poset Dimension Number, $\mathsf{dim}(G)$[5])**.** *Let $\vec{P}$ be any poset. We say that a mapping $\varphi : V(\vec{P}) \to \mathbb{R}^d$* realizes *poset $\vec{P}$ if for any distinct vertices $u, v \in V(\vec{P})$, $uv \in E(\vec{P})$ if and only if $\varphi(u) < \varphi(v)$. The dimension of poset $\vec{P}$, denoted by $\mathsf{dim}(\vec{P})$, is the smallest integer d such that there is a mapping $\varphi : V(\vec{P}) \to \mathbb{R}^d$ that realizes $\vec{P}$.*

Why Height-two Posets? Recall that our main theorem only applies to height-two posets. There is a good reason for this. Note that it is not always the case that the product $G \times_e \vec{P}$ between an undirected graph G and poset $\vec{P}$ (that we use in Theorem 1.1) will result in a poset (an example is when G and $\vec{P}$ are a path and a directed path of three vertices, respectively); so, the term $\mathsf{dim}(G \times_e \vec{P})$ does not necessarily make sense. However, if $\vec{P}$ is a *height-two* poset, then $G \times_e \vec{P}$ is always a poset (in fact, a height-two one). (We prove this fact in Lemma 4.8 in Section 4.) For this reason, Theorem 1.1 is stated only for a height-two poset $\vec{P}$.

3 Proof of the Special case of Theorem 1.1

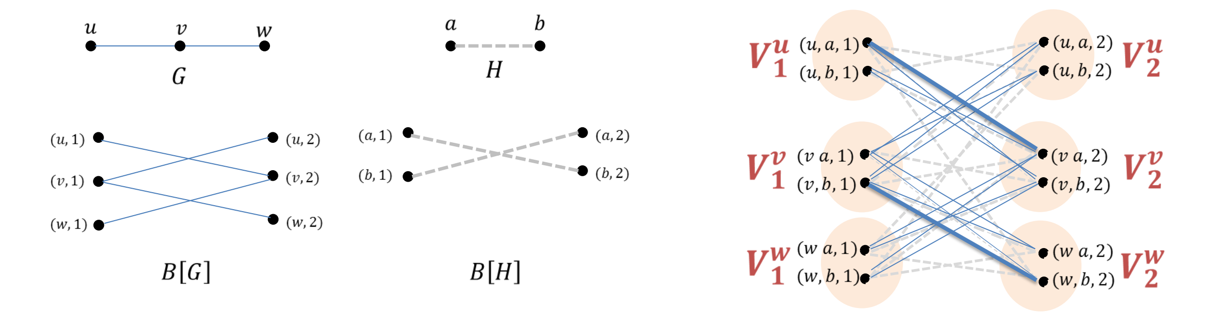
In this section, we will focus on the special case of Theorem 1.1 where we consider the products $(G \oplus H) * J$ when $J = K_2$ only. This is partly because the proof is more intuitive (and easier to illustrate by pictures) in this special case. Moreover, this case itself is sufficient for our purpose in proving hardness results. We will also use $B[G \oplus H]$ instead of $(G \oplus H) * K_2$ to simplify the notation (more on this in Section 3.1). Proofs of the general cases can be found in Section 4. They are relatively short, perhaps easier to verify, and can be read without understanding any material in this section. However, since the proofs in the general case are less intuitive, some readers might find the intuition in this section helpful.

3.1 Why Multiplying Graphs by K_2?

We first give a motivation of studying the graph products in this specific (and rather peculiar) form. First, notice that both $G \times K_2$ and $G \times_e K_2$ are *bipartite*. To see this, let $V(K_2) = \{1, 2\}$; so,

[4] Our definition follows [31].

[5] This concept is also sometime called the *order dimension* or the *Dushnik–Miller dimension*. It is usually defined using the notion of *realizer*. This is however equivalent to our "embedding to $\mathbb{R}^d$" definition, and we use the embedding definition as it will be easier to use in our proofs.



(a) G, H, B$[G]$ and B$[H]$

(b) B$[G \vee H]$, super vertices V_i^x and induced matching $\mathcal{M}_G$

Figure 2: Example of graphs G, H, B$[G]$, B$[H]$ and B$[G \vee H]$, as well as super vertices V_i^x, set of edges E_G and induced matching $\mathcal{M}_G$ (defined in Section 3.2). Bold edges are in $\mathcal{M}_G$. Solid edges (in blue, including bold edges) are edges assigned to E_G, and dashed edges (in gray) are edges assigned to E_H. Observe that if we view V_i^x as a vertex (by unifying vertices in them) and consider only edges in E_G, then the graph looks exactly like B$[G]$. Moreover, the induced matching $\mathcal{M}_G$ becomes an induced matching $\{V_1^u V_2^v, V_1^v V_2^w\}$ in this graph of super vertices. This is the main fact we use to prove Eq.(5).

the vertices of $G \times K_2$ and $G \times_e K_2$ are in the form (v, i) where $v \in V(G)$ and $i \in \{1, 2\}$ (e.g., Fig. 2 and 4). Thus, we can partition vertices of these graphs into $V_1 \cup V_2$ where $V_i = \{(v, i) : v \in V(G)\}$. There is no edge between vertices in the same partition in both graphs so they are bipartite.

So, we may think of the product $G \times K_2$ and $G \times_e K_2$ as a "transformation" of graph G into a bipartite graph. To emphasize this point of view and simplify the notation later on, we will write B$[G]$ and B$_e[G]$ instead of $G \times K_2$ and $G \times_e K_2$ in this and next sections.

An intuitive way to think of B$[G]$ and B$_e[G]$ is to imagine the following construction. We construct B$[G]$ by making two copies of each vertex $v \in V(G)$ to get vertices $(v, 1)$ and $(v, 2)$ in B$[G]$. Then, for each edge $uv \in E(G)$, we add an edge between $(u, 1)$ and $(v, 2)$. To get B$_e[G]$, we simply add an edge connecting the two copies of each vertex $v \in V(G)$; i.e., we add an edge between $(v, 1)$ and $(v, 2)$ for all $v \in V(G)$.

This transformation might now sound familiar to many readers. The graph B$[G]$ is actually known as the *bipartite double cover* and has been used repeatedly as a natural way to transform any graph into a bipartite graph; for example, one can use this transformation to reduce the problem of computing cycle covers to the maximum bipartite matching problem.

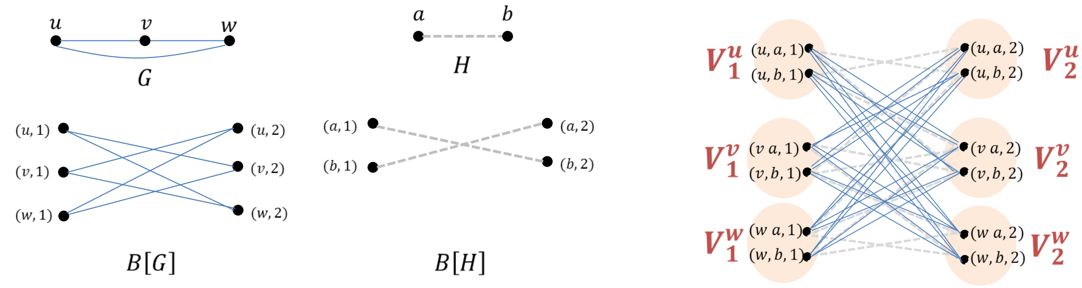
In the context of posets, we will abuse the notation and define B$[G] = G \times \vec{K}_2$ and B$_e[G] = G \times_e \vec{K}_2$. Edges in these graphs always point from vertices in V_1 to vertices in V_2. The graph B$[G]$ also has its name in this case – an *adjacency poset* [20].

3.2 Induced Matching Number (Eq.(1)).

In this section, we aim to prove the following special case of Eq.(1):

$$\mathsf{im}(\mathrm{B}[G \vee H]) \leq \mathsf{im}(\mathrm{B}[G]) + \mathsf{im}(\mathrm{B}[H]). \tag{5}$$

Let V_1 and V_2 be the two partitions of vertices in B$[G \vee H]$ and $\mathcal{M}$ be an induced matching in B$[G \vee H]$. Recall that each edge in B$[G \vee H]$ is of the form $(u, a, 1)(v, b, 2)$, where $u, v \in V(G)$ and $a, b \in V(H)$, and it appears in B$[G \vee H]$ if and only if at least one of the following conditions holds: (1) $uv \in E(G)$ or (2) $ab \in E(H)$. Our strategy is to consider edges satisfying each condition separately.



(a) G, H, B$[G]$ and B$[H]$ (b) B$[G \vee H]$

Figure 3: Example of graphs G, H, B$[G]$, B$[H]$ and B$[G \vee H]$, as well as super vertices V_i^x. Solid edges (in blue) are edges assigned to E_G, and dashed edges (in gray) are edges assigned to E_H. (E_G and E_H are defined in Section 3.2.)

In particular, we let $E(\mathrm{B}[G \vee H]) = E_G \cup E_H$, where E_G and E_H consist of edges $(u, a, 1)(v, b, 2)$ that satisfy the first and second condition, respectively. That is, $E_G = \{(u, a, 1)(v, b, 2) : uv \in E(G)\}$ and $E_H = \{(u, a, 1)(v, b, 2) : ab \in E(H)\}$. For example, in Fig. 2(b), E_G consists of solid edges (in blue) and E_H consists of dashed edges (in gray). Note that some edges, e.g., edge $(u, a, 1)(v, b, 2)$ in Fig. 2(b), are in both E_G and E_H. We also partition the induced matching $\mathcal{M}$ into $\mathcal{M} = \mathcal{M}_G \cup \mathcal{M}_H$ where $\mathcal{M}_G = \mathcal{M} \cap E_G$ and $\mathcal{M}_H = \mathcal{M} \cap E_H$. Obviously, $|\mathcal{M}| \leq |\mathcal{M}_G| + |\mathcal{M}_H|$. Our goal is to show that $|\mathcal{M}_G| \leq \mathsf{im}(\mathrm{B}[G])$ and $|\mathcal{M}_H| \leq \mathsf{im}(\mathrm{B}[H])$. We will only show the former claim because the latter can be argued similarly.

To prove this claim, we partition vertices in V_1 and V_2 according to which vertices in G they "inherit" from. That is, for any vertex $u \in V(G)$, we let $V_1^u = \{(u, a, 1) : a \in V(H)\}$ and $V_2^u = \{(u, a, 2) : a \in V(H)\}$ (e.g., see Fig. 2(b)).

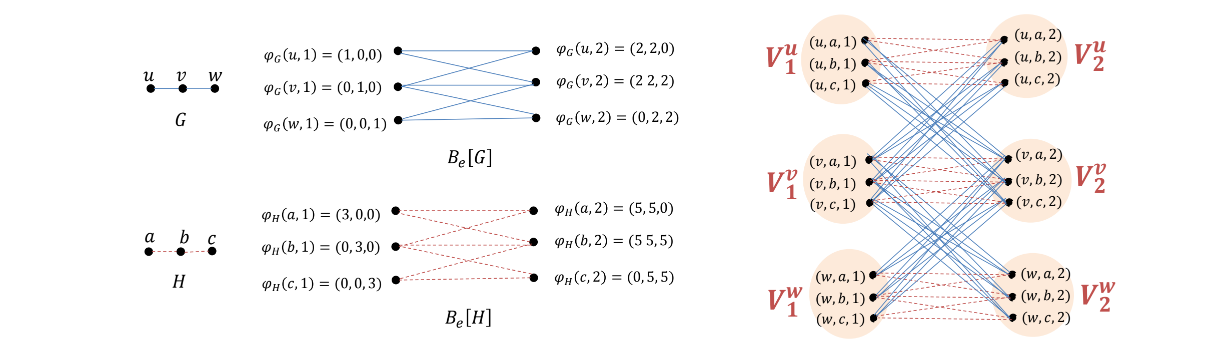
We can think of each set V_i^u as a "super vertex" corresponding to a vertex (u, i) in B$[G]$ in the sense that if we unify all vertices in V_i^u into one vertex, for all $u \in V(G)$ and $i \in V(K_2)$, and remove duplicate edges, then we will get the graph B$[G]$. In fact, we can show more than this. We can show that if we look at $\mathcal{M}_G$ in the graph of super vertices, then we will get an induced matching of B$[G]$ having the same size as $\mathcal{M}_G$! For example, in Fig. 2(b) the induced matching $\mathcal{M}_G$ in B$[G \vee H]$ consisting of bold edges becomes a set of two edges $\{V_1^u V_2^v, V_1^v V_2^w\}$ in the graph of super vertices, which is still an induced matching.

The key idea in proving this fact is an observation that for any pair of super vertices V_1^u and V_2^v, either there is no edge between any pair of vertices in V_1^u and V_2^v, or there will be edges between all pairs of vertices in V_1^u and V_2^v. For example, in Fig. 2(b), there is no edge between any pair of vertices $x \in V_1^u$ and $y \in V_2^w$ while there is an edge between every pair of vertices $x \in V_1^u$ and $y \in V_2^v$. Using this observation, we can easily prove the two lemmas below. The first lemma says that $\mathcal{M}_G$ becomes a matching in the graph of super vertices, and the second one says that this matching is, in fact, an induced matching.

Before proceeding to the proofs, recall that we write the edge set of B$[G \vee H]$ as $E(\mathrm{B}[G \vee H]) = E_G \cup E_H$, where $E_G = \{(u, a, 1)(v, b, 2) : uv \in E(G)\}$ and $E_H = \{(u, a, 1)(v, b, 2) : ab \in E(H)\}$.

Lemma 3.1. *For any $u \in V(G)$ and $i \in \{1, 2\}$, V_i^u contains an endpoint of at most one edge in $\mathcal{M}_G$.*

Proof. For the sake of contradiction, assume that there is a vertex $u \in V(G)$ such that V_1^u contains two endpoints of two edges in $\mathcal{M}_G$, say $(u, a, 1)(v, b, 2)$ and $(u, a', 1)(v', b', 2)$. (The case of V_2^u is proved analogously.) Since $(u, a, 1)(v, b, 2)$ is in E_G (recall that $\mathcal{M}_G = \mathcal{M} \cap E_G$), we have that



(a) G, H, $\mathrm{B}_e[G]$ and $\mathrm{B}_e[H]$ (b) $\mathrm{B}_e[G \cdot H]$

Figure 4: Graphs G, H, $\mathrm{B}_e[G]$, $\mathrm{B}_e[H]$ and $\mathrm{B}_e[G \cdot H]$, as well as super vertices V_i^x and mappings φ_G and φ_H that realize $\mathrm{B}_e[G]$ and $\mathrm{B}_e[H]$, respectively. Note that directions of edges are omitted in the pictures. They are always from left to right.

$uv \in E(G)$ and thus $(u, 1)(v, 2)$ is in $E(\mathrm{B}[G])$. This fact then implies that there is an edge between $(u, a', 1)$ and $(v, b, 2)$ in E_G as well, contradicting to the fact that $\mathcal{M}_G$ (and thus $\mathcal{M}$) is an induced matching. $\square$

Example. Here we illustrate the proof of Lemma 3.1. Consider Fig. 3(b) and let us say that $\mathcal{M}_G$ contains edges $(u, a, 1)(v, b, 2)$ and $(u, b, 1)(v, a, 2)$ which means that V_1^u contains endpoints of two edges in $\mathcal{M}_G$. Having the first edge in E_G means that $uv \in E(G)$ and thus $(u, 1)(v, 2)$ is in $E(\mathrm{B}[G])$ (as witnessed in Fig. 3(a)). But then it means that the edge $(u, a, 1)(v, a, 2)$ must be in E_G as well, making $\mathcal{M}_G$ (and thus $\mathcal{M}$) not an induced matching.

Lemma 3.2. *For any $u, u', v, v' \in V(G)$, if $\mathcal{M}_G$ contains an edge between a pair of vertices in V_1^u and V_2^v and an edge between another pair of vertices in $V_1^{u'}$ and $V_2^{v'}$, then there must be no edge between vertices in V_1^u and $V_2^{v'}$ in E_G.*

Proof. Assume for a contradiction that $\mathcal{M}_G$ contains edges $(u, a, 1)(v, b, 2)$ and $(u', a', 1)(v', b', 2)$ and there is an edge, say $(u, c, 1)(v', d, 2)$ in $E(G)$. Since the edge $(u, c, 1)(v', d, 2)$ is in E_G, we have $uv' \in E(G)$ and thus $(u, 1)(v', 2) \in E(\mathrm{B}[G])$. This implies that $(u, a, 1)(v', b', 2)$ is in E_G, which contradicts the fact that $\mathcal{M}$ is an induced matching in $\mathrm{B}[G \vee H]$. $\square$

Example. Here we illustrate the proof of Lemma 3.2. Consider Fig. 3(b), and let us say that the matching $\mathcal{M}_G$ contains $(v, a, 1)(u, a, 2)$ and $(u, a, 1)(w, a, 2)$ and there is an edge $(v, b, 1)(w, b, 2)$ which prevents $\mathcal{M}_G$ from being an induced matching in the graph of super vertices. Having the last edge in E_G implies that $(v, 1)(w, 2) \in E(\mathrm{B}[G])$ which in turns implies that $(v, a, 1)(w, a, 2) \in E_G$, making $\mathcal{M}_G$ (and thus $\mathcal{M}$) not an induced matching in $\mathrm{B}[G \vee H]$.

Note on Proving the General Version (Section 4). To prove the general version, i.e., Eq.(1), we may define E_G and E_H analogously to the proof in this section. We may then define super vertices in a similar way and prove the lemmas that are similar in spirit to Lemma 3.1 and 3.2. However, we choose an *alternative way* which seems more suitable in proving the general version by decomposing the graph into products of some well-structured graphs and show the *associativity* property of graph products we use.

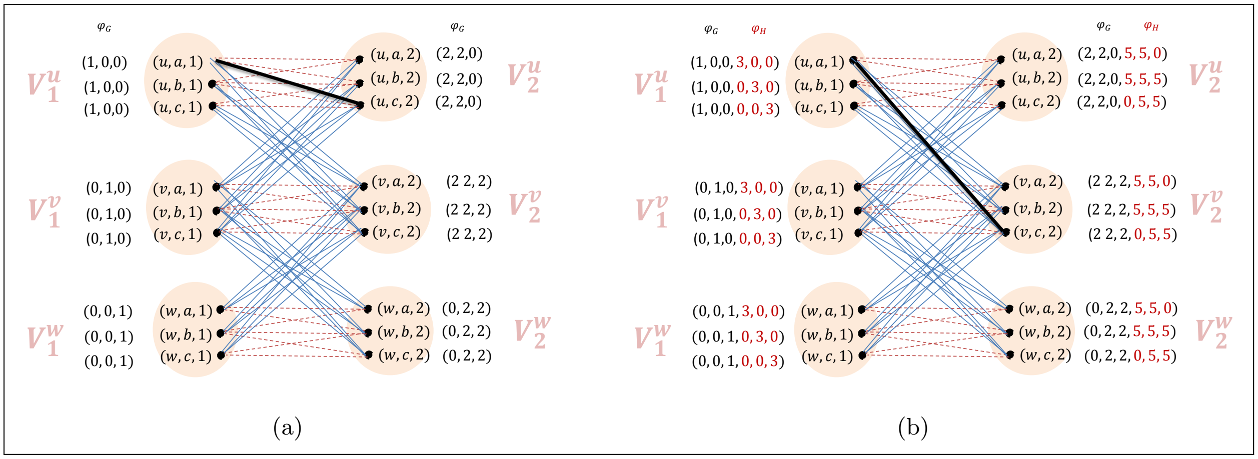


Figure 5: (a) φ_1 (cf. Section 3) introduces "undesirable edges" such as the bold edge in this picture. (b) φ_2 (cf. Section 3) removes "desirable edges" such as the bold edge in this picture.

3.3 Poset Dimension Number (Eq.(3)).

We now prove the special case of Eq. (3)[6]:

$$\mathsf{dim}(\mathrm{B}_e[G \cdot H]) \le \mathsf{dim}(\mathrm{B}_e[G]) + \chi(G)\mathsf{dim}(\mathrm{B}_e[H])\,.$$

Throughout this section, we will think of $\mathrm{B}_e[G]$, for any undirected graph G, as a poset $G \times_e \vec{K}_2$. Thus, edges in $\mathrm{B}_e[G \cdot H]$ are directed edges in the form $(u, a, 1)(v, b, 2)$ for some $u, v \in V(G)$ and $a, b \in V(H)$. Let $d_G = \mathsf{dim}(\mathrm{B}_e[G])$, $d_H = \mathsf{dim}(\mathrm{B}_e[H])$, and $\varphi_G : V(\mathrm{B}_e[G]) \to \mathbb{R}^{d_G}, \varphi_H : V(\mathrm{B}_e[H]) \to \mathbb{R}^{d_H}$ be mappings that realize the posets $\mathrm{B}_e[G]$ and $\mathrm{B}_e[H]$, respectively. This means that, for example, $(u, 1)(v, 2) \in E(\mathrm{B}_e[G])$ if and only if $\varphi_G(u, 1) < \varphi_G(v, 2)$. We may assume without loss of generality that all coordinates of φ_G and φ_H are non-negative (by adding appropriate positive numbers). See an example in Fig. 4.

Our strategy is to use φ_G and φ_H to define a mapping $\varphi : V(\mathrm{B}_e[G \cdot H]) \to \mathbb{R}^{d_G+\chi(G)d_H}$ that realizes $\mathrm{B}_e[G \cdot H]$. Again, this means that we want φ such that for any vertices (u, a, i) and (v, b, j), $(u, a, i)(v, b, j) \in E(\mathrm{B}_e[G \cdot H])$ if and only if $\varphi(u, a, i) < \varphi(v, b, j)$. To simplify our discussion, we will focus on the case where $i = 1$ and $j = 2$ (The cases when $i = j$ are easy to deal with).

Proof Idea. Before we show the construction of φ, let us show a few *failed attempts* to illustrate the intuition behind the construction (readers may feel free to skip this part to the definition of φ below). We use Fig. 4 as an example. Fig. 5 and 7 might be also helpful as visual aids.

The first attempt is to use $\varphi_1(u, a, i) = \varphi_G(u, i)$ to realize $\mathrm{B}_e[G \cdot H]$. This obviously fails, simply because we did not use φ_H at all: In Fig. 4 (also see Fig. 5(a)), we have $\varphi_1(u, a, 1) < \varphi_1(u, c, 2)$, but $(u, a, 1)(u, c, 2) \notin E(\mathrm{B}_e[G \cdot H])$. In other words, φ_1 "introduces" some "undesirable edges" – edges $(u', a', 1)(v', b', 2)$ that are *not* in $\mathrm{B}_e[G \cdot H]$ but $\varphi_1(u', a', 1) < \varphi_1(v', b', 2)$.

A natural way to fix this is to define $\varphi_2(u, a, i) = \varphi_G(u, i)\varphi_H(a, i)$ which is a "concatenation" of $\varphi_G(u, i)$ and $\varphi_H(a, i)$ (thus, the dimension of φ_2 is $d_G + d_H$). It can be shown that there is no undesirable edge introduced by φ_2. However, φ_2 might "remove" some "desirable edges" – edges $(u', a', 1)(v', b', 2)$ that *are* in $\mathrm{B}_e[G \cdot H]$ but $\varphi_1(u', a', 1) \not< \varphi_1(v', b', 2)$. For example, in Fig. 4 (also see Fig. 5(b)), $\varphi_2(u, a, 1) \not< \varphi_2(v, c, 2)$, but $(u, a, 1)(v, c, 2) \in E(\mathrm{B}_e[G \cdot H])$.

[6]Note that Eq.(3) implies that $\mathsf{dim}(\mathrm{B}_e[G \cdot H]) \le \mathsf{dim}(\mathrm{B}_e[G]) + \chi(G)\mathsf{dim}(\mathrm{B}_e[H]) + 1$, so we are proving a slightly stronger statement for this special case.

$$\varphi_3(x,r,1) = \boxed{\varphi_G(x,1)}\ \boxed{\varphi_H(r,1)}\ \boxed{(0,0,0)} \quad \text{and} \quad \varphi_3(x,r,2) = \boxed{\varphi_G(x,2)}\ \boxed{\varphi_H(r,2)}\ \boxed{(\infty,\infty,\infty)}$$
$$\varphi_3(v,r,1) = \boxed{\varphi_G(v,1)}\ \boxed{(0,0,0)}\ \boxed{\varphi_H(r,1)} \quad \text{and} \quad \varphi_3(v,r,1) = \boxed{\varphi_G(v,2)}\ \boxed{(\infty,\infty,\infty)}\ \boxed{\varphi_H(r,2)}$$

Figure 6: Example of φ_3. Note that x is node in $\{u,w\}$.

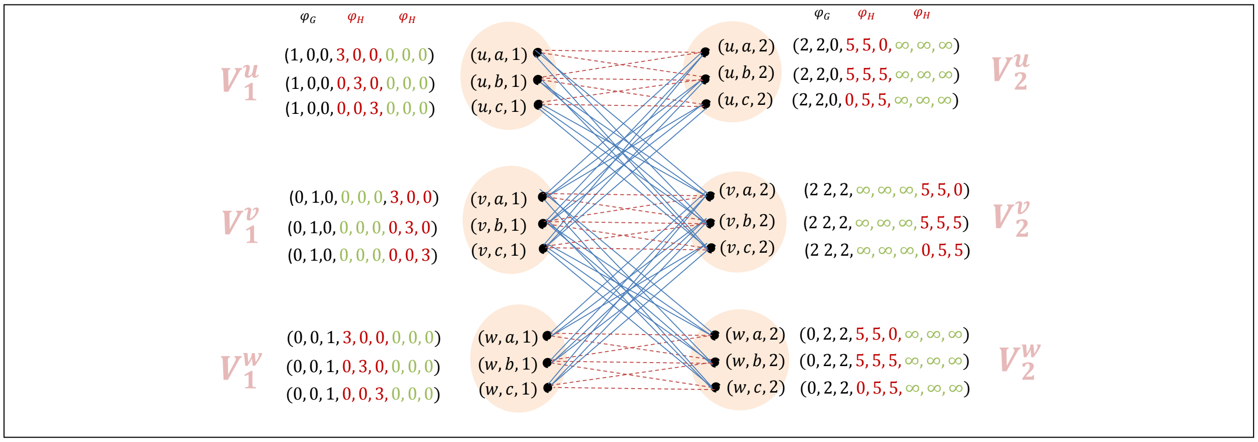


Figure 7: φ_3 (cf. Section 3) which realizes $\mathrm{B}_e[G \cdot H]$.

We thus need a more clever way to combine φ_G with φ_H. A crucial observation we found is that *if we concatenate them only at vertices that are independent in G, then we will not remove any desirable edges.* For example, in Fig. 4(a), vertices u and w are independent in G. So, for any $x \in \{u,w\}$ and $r \in V(H)$, we will let φ_3 be as in Fig. 6. In other words, every vertex starts with φ_G (i.e., in the first (gray) boxes). Then, we "attach" φ_H to φ_G at vertices of the form (x,r,i), where $x \in \{u,w\}$, while attaching "trivial" vectors ($(0,0,0)$ or (∞,∞,∞)) at (v,r,i) (i.e., in the second (blue) boxes in Fig 6). We then do the opposite. We attach φ_H to vertices (v,r,i) while attaching trivial vectors to other vertices (i.e., in the last (green) boxes). It can be checked (e.g., Fig. 7) that φ_3 does realize $\mathrm{B}_e[G \cdot H]$.

To summarize, the general idea of constructing φ is to keep attaching φ_H to φ_G where each attachment must be done only on vertices that are *independent in G*. The dimension of φ depends on how many times we attach φ_H. A natural way to minimize the number of attachments is to use $\chi(G)$ color classes since each color class contains independent vertices. This is why the dimension becomes $d_G + \chi(G)d_H$.

Constructing φ. Let $\mathcal{C} : V(G) \to [k]$ be an optimal coloring of G, where $k = \chi(G)$. The coordinates in $\mathbb{R}^{d_G + kd_H}$ are viewed as $k+1$ *blocks*. In the first block $\mathcal{B}_0$, we have $d_G = \mathsf{dim}(\mathrm{B}_e[G])$ coordinates, and in the k consecutive blocks $\mathcal{B}_1, \ldots, \mathcal{B}_k$, we have $d_H = \mathsf{dim}(\mathrm{B}_e[H])$ coordinates per block. We will define the coordinates of each vertex in $\mathrm{B}_e[G \cdot H]$ by describing the coordinates in each block. For each point $\mathbf{x} \in \mathbb{R}^{d_G + kd_H}$ and each block $\mathcal{B}_j$, we refer to coordinates in block $\mathcal{B}_j$ of $\mathbf{x}$ as $\mathbf{x}|_{\mathcal{B}_j}$.

For each vertex in $\mathrm{B}_e[G \cdot H]$ of the form $(u,a,1)$, we define its coordinates as

$$\begin{aligned}
\varphi((u,a,1))|_{\mathcal{B}_0} &= \varphi_G((u,1)), & \\
\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(u)}} &= \varphi_H((a,1)), & \text{and} \\
\varphi((u,a,1))|_{\mathcal{B}_\ell} &= (0,\ldots,0), & \text{otherwise.}
\end{aligned}$$

For each vertex in $\mathrm{B}_e[G \cdot H]$ of the form $(u, a, 2)$, we define its coordinates as

$$
\begin{array}{lll}
\varphi((u,a,2))|_{\mathcal{B}_0} & = \varphi_G((u,2)), & \\
\varphi((u,a,2))|_{\mathcal{B}_{\mathcal{C}(u)}} & = \varphi_H((a,2)), & \text{and} \\
\varphi((u,a,2))|_{\mathcal{B}_\ell} & = (\infty,\dots,\infty), & \text{otherwise.}
\end{array}
$$

We finish the proof of Eq.(6) by the following lemma, which can be proved by case analysis.

Lemma 3.3. *φ realizes $B_e[G \cdot H]$.*

Proof. We will use the following properties:

P1. For any $(u,1),(v,2) \in V(\mathrm{B}_e[G])$, $(u,1)(v,2) \in E(\mathrm{B}_e[G]) \iff \varphi_G((u,1)) < \varphi_G((v,2))$.

P2. For any $(a,1)(b,2) \in V(\mathrm{B}_e[H])$, $(a,1)(b,2) \in E(\mathrm{B}_e[H]) \iff \varphi_H((a,1)) < \varphi_H((b,2))$.

P3. For any $uv \in E(G)$, $\mathcal{C}(u) \neq \mathcal{C}(v)$.

We argue that for any vertices $(u,a,1),(v,b,2) \in V(\mathrm{B}_e[G \cdot H])$, we have $(u,a,1)(v,b,2) \in E(\mathrm{B}_e[G \cdot H])$ if and only if $\varphi((u,a,1)) < \varphi((v,b,2))$. We analyze five possible cases.

Case 1: $u \neq v$ and $uv \in E(G)$. We will show that $\varphi((u,a,1)) < \varphi((v,b,2))$. First, $(u,a,1)(v,b,2) \in E(\mathrm{B}_e[G \cdot H])$ by construction, and $\mathcal{C}(u) \neq \mathcal{C}(v)$ by Property P3. Next, consider the blocks $\mathcal{B}_0$, $\mathcal{B}_{\mathcal{C}(u)}$ and $\mathcal{B}_{\mathcal{C}(v)}$. We have

$$
\begin{aligned}
\varphi((u,a,1))|_{\mathcal{B}_0} &= \varphi_G((u,1)) < \varphi_G((v,2)) = \varphi((v,b,2))|_{\mathcal{B}_0} \\
\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(u)}} &= \varphi_H((a,1)) < \infty = \varphi((v,b,2))|_{\mathcal{B}(\mathcal{C}(u))} \\
\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(v)}} &= 0 < \varphi_G((v,2)) = \varphi((v,b,2))|_{\mathcal{B}_{\mathcal{C}(v)}}
\end{aligned}
$$

The first line is because of Property P1 and the fact that $(u,1)(v,2) \in E(\mathrm{B}_e[G])$. This proves the claim.

Case 2: $u \neq v$ and $uv \notin E(G)$. We will show that $\varphi((u,a,1)) \not< \varphi((v,b,2))$. First, $(u,a,1)(v,b,2) \notin E(\mathrm{B}_e[G \cdot H])$ by construction. Consider the block $\mathcal{B}_0$. Because of Property P1 and $(u,1)(v,2) \notin E(\mathrm{B}_e[G])$, we have $\varphi((u,a,1))|_{\mathcal{B}_0} = \varphi_G((u,1)) \not< \varphi_G((v,2)) = \varphi((v,b,2))|_{\mathcal{B}_0}$. Thus, $\varphi((u,a,1)) \not< \varphi((v,b,2))$.

Case 3: $u = v$, $a \neq b$ and $ab \in E(H)$. We will show that $\varphi((u,a,1)) < \varphi((v,b,2))$. First, $(u,a,1)(v,b,2) \in E(\mathrm{B}_e[G \cdot H])$ by construction. Note that $\mathcal{C}(u) = \mathcal{C}(v)$ since $u = v$. Consider each block. We have

$$
\varphi((u,a,1))|_{\mathcal{B}_0} = \varphi_G((u,1)) = \varphi_G((v,2)) = \varphi((v,b,2))|_{\mathcal{B}_0} \tag{6}
$$
$$
\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(u)}} = \varphi_H((a,1)) < \varphi_H((b,2)) = \varphi((v,b,2))|_{\mathcal{B}_{\mathcal{C}(v)}} \tag{7}
$$
$$
\varphi((u,a,1))|_{\mathcal{B}_\ell} = 0 \leq \infty = \varphi((v,b,2))|_{\mathcal{B}_\ell} \tag{8}
$$

Eq.(7) follows because $(a,1)(b,2) \in E(\mathrm{B}_e[H])$ and Property P2, and Eq.(8) follows from the settings of other blocks $\mathcal{B}_\ell$. This proves the claim.

Case 4: $u = v$, $a \neq b$ and $ab \notin E(H)$. We will show that $\varphi((u,a,1)) \not< \varphi((v,b,2))$. First, $(u,a,1)(v,b,2) \notin E(\mathrm{B}_e[G \cdot H])$ by construction. Consider the block $\mathcal{B}_{\mathcal{C}(u)}$. By Property P2, $\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(u)}} = \varphi_H((a,1)) \not< \varphi_H((b,2)) = \varphi((v,b,2))|_{\mathcal{B}_{\mathcal{C}(v)}}$, thus proving the claim.

Case 5: $u = v$ **and** $a = b$**.** We will show that $\varphi((u,a,1)) < \varphi((v,b,2))$. First, by the definition of the operator B_e, $(u,a,1)(v,b,2) \in E(\mathrm{B}_e[G \cdot H])$. Next, consider each block.

$$\varphi((u,a,1))|_{\mathcal{B}_0} = \varphi_G((u,1)) < \varphi_G((v,2)) = \varphi((v,b,2))|_{\mathcal{B}_0} \tag{9}$$
$$\varphi((u,a,1))|_{\mathcal{B}_{\mathcal{C}(u)}} = \varphi_H((a,1)) < \varphi_H((b,2)) = \varphi((v,b,2))|_{\mathcal{B}_{\mathcal{C}(v)}} \tag{10}$$
$$\varphi((u,a,1))|_{\mathcal{B}_\ell} = 0 \leq \infty = \varphi((v,b,2))|_{\mathcal{B}_\ell} \tag{11}$$

Eq.(10) follows from Property P2, and Eq.(11) follows from the settings of other blocks $\mathcal{B}_\ell$. This proves the claim. □

4 Proof of the General Case of Theorem 1.1

In this section, we prove the main theorem. We shall restate the main theorem here.

Theorem 1.1.(restated) *For any undirected graphs G, H and J and a height-two poset $\vec{P}$,*

$$\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(G \times J) + \mathsf{im}(H \times J) \tag{1}$$
$$\mathsf{sim}((G \vee H) \times J) \leq \mathsf{sim}(G \times J) + \mathsf{sim}(H \times J) \tag{2}$$
$$\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}(G \times_e \vec{P}) + \chi(G)\mathsf{dim}(H \times_e \vec{P}) + \mathsf{dim}(\vec{P}) \tag{3}$$

where $\chi(G)$ is the chromatic number of G.

4.1 Subadditivity of Induced Matching Number (Eq.(1)).

In this section, we prove that $\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(G \times J) + \mathsf{im}(H \times J)$. We first observe that we can decompose edges of $G \vee H$ into two sets:

$$E(G \vee H) = \{(u,a)(v,b) \mid uv \in E(G) \text{ or } ab \in E(H)\} = E_1 \cup E_2 \tag{12}$$

where $E_1 = \{(u,a)(v,b) \in E(G \vee H) : uv \in E(G)\}$ and $E_2 = \{(u,a)(v,b) \in E(G \vee H) : ab \in E(H)\}$. For any $i \in \{1,2\}$, define a subgraph G_i of $G \vee H$ to be $G_i = (V(G \vee H), E_i)$. Note that $E((G \vee H) \times J) \subseteq E(G_1 \times J) \cup E(G_2 \times J)$.

Claim 4.1. $\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(G_1 \times J) + \mathsf{im}(G_2 \times J)$.

Proof. Let $\mathcal{M}$ be any induced matching in the graph $(G \vee H) \times J$. Let $\mathcal{M}_1 = \mathcal{M} \cap E(G_1 \times J)$ and $\mathcal{M}_2 = \mathcal{M} \cap E(G_2 \times J)$. By Eq.(12), $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$. Observe that $\mathcal{M}_1$ and $\mathcal{M}_2$ are induced matchings of G_1 and G_2, respectively, since they are induced matchings of $(G \vee H) \times J$ which is a super graph of $G_1 \times J$ and $G_2 \times J$. The claim follows.

□

Now, we try to write G_1 and G_2 as a product of two other graphs. For any set X of vertices, we denote by K_X a complete graph whose vertex set is $V(X)$.

Lemma 4.2. $\mathsf{im}(G_1 \times J) = \mathsf{im}((K_H \times_e G) \times J)$ *and* $\mathsf{im}(G_2 \times J) = \mathsf{im}((K_G \times_e H) \times J)$.

Proof. The lemma simply follows from the fact that $K_G \times_e H$ is exactly the same as G_2 and $K_H \times_e G$ is isomorphic to G_1. To see this, we simply observe that $E(K_G \times_e H) = \{(u,a)(v,b) : ab \in E(H) \wedge u,v \in V(G)\}$ which is exactly the same as E_2, and $E(K_H \times_e G) = \{(a,u)(b,v) : uv \in E(G) \wedge a,b \in V(H)\}$ which is almost the same as E_1 except that vertices are in $V(H) \times V(G)$ instead of $V(G) \times V(H)$.

□

The simple lemma above allows us to rewrite the equation in Claim 4.1 as

$$\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}((K_H \times_e G) \times J) + \mathsf{im}((K_G \times_e H) \times J) \tag{13}$$

Now we need the following *associativity* property.

Lemma 4.3. *For any graphs X, Y and Z, $(X \times_e Y) \times Z = X \times_e (Y \times Z)$.*

Proof. The following equalities simply follow from the definition of $\times$ and $\times_e$ (cf. Definition 2.1).

$$\begin{aligned} E((X \times_e Y) \times Z) &= \{(x,y,z)(x',y',z') : (x,y)(x',y') \in E(X \times_e Y) \text{ and } zz' \in E(Z)\} \\ &= \{(x,y,z)(x',y',z') : (xx' \in E(X) \text{ or } x = x') \text{ and } yy' \in E(Y) \text{ and } zz' \in E(Z)\} \\ &= X \times_e (Y \times Z) \end{aligned}$$

□

This allows us to rewrite Eq.(13) as

$$\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(K_H \times_e (G \times J)) + \mathsf{im}(K_G \times_e (H \times J)) \tag{14}$$

We finish our proof with the following lemma which says that the product of any graph X with a complete graph K_L will not increase the induced matching number of X. We note that in fact the equality could be achieved, but since it is not important to this proof, we only show the upper bound.

Lemma 4.4. *For any graph X and any set L of vertices, $\mathsf{im}(K_L \times_e X) \leq \mathsf{im}(X)$.*

Proof. Let $\mathcal{M}$ be any induced matching in $K_L \times_e X$. We construct a set of edges $\mathcal{M}_X \subseteq E(X)$ by projection: for each edge $(i,x)(j,y) \in E(K_L \times_e X)$, we add an edge xy to $\mathcal{M}_X$. To prove the lemma, it suffices to show that $\mathcal{M}_X$ is an induced matching in graph X.

Assume for contrary that $\mathcal{M}_X$ is not an induced matching, i.e., there exists $xy, x'y' \in \mathcal{M}_X$ such that either (1) $x = x'$ (making $\mathcal{M}_X$ not a matching) or (2) $xx' \in E(X)$ (making $\mathcal{M}_X$ not an induced matching). (We note that (1) also includes the case where multiple edges are created, i.e., $x = x'$ and $y' = y'$.) We shall use the following simple facts.

$$xy \in \mathcal{M}_X \implies \exists i,j : (i,x)(j,y) \in \mathcal{M} \tag{15}$$

$$xy \in E(X) \implies \forall i,j : (i,x)(j,y) \in E(K_L \times_e X) \tag{16}$$

By Eq.(15), the assumption that $xy, x'y' \in \mathcal{M}_X$ implies that edges $(i,x)(j,y)$ and $(i',x')(j',y')$ belong to $\mathcal{M}$ for some $i,j,i',j' \in V(K_L)$.

Case 1: If $x = x'$, then we have that xy' is also in $\mathcal{M}_X \subseteq E(X)$ and thus Eq.(16) implies that $(i,x)(j',y') \in E(K_L \times_e X)$. This contradicts the fact that $\mathcal{M}$ is an induced matching.

Case 2: If $xx' \in E(X)$, then Eq.(16) implies that $(i,x)(i',x') \in E(K_L \times_e X)$ which again contradicts the fact that $\mathcal{M}$ is an induced matching.

□

Using Lemma 4.4, we can rewrite Eq.(14) as $\mathsf{im}((G \vee H) \times J) \leq \mathsf{im}(G \times J) + \mathsf{im}(H \times J)$ as desired.

4.2 Subadditivity of Semi-induced Matching Number (Eq.(2)).

In this section, we prove the subadditivity property of the semi-induced matching number. The proof closely follows the case of the induced matching number.

We prove the following subadditivity theorem for semi-induced matching which is equivalent to Eq.(2).

Theorem 4.5. *For any graphs G and H and any total order σ on $V((G \vee H) \times J)$, there exist bijections σ_1 on $V(G \times J)$ and σ_2 on $V(H \times J)$ such that*

$$\mathsf{sim}_\sigma((G \vee H) \times J) \leq \mathsf{sim}_{\sigma_1}(G \times J) + \mathsf{sim}_{\sigma_2}(H \times J)\,.$$

The rest of this subsection is devoted to proving the above theorem. We first decompose edge set $E(G \vee H)$ into $E_1 \cup E_2$ where $E_1 = \{(u,a)(v,b) : uv \in E(G) \ \wedge \ a,b \in V(H)\}$ and $E_2 = \{(u,a)(v,b) : ab \in E(H) \ \wedge \ u,v \in V(G)\}$. For any $i \in \{1,2\}$, define a subgraph G_i of $G \vee H$ to be $G_i = (V(G \vee H), E_i)$.

Claim 4.6. *For any bijection $\sigma : V((G \vee H) \times J) \to [|V((G \vee H) \times J)|]$, $\mathsf{sim}_\sigma((G \vee H) \times J) \leq \mathsf{sim}_\sigma(G_1 \times J) + \mathsf{sim}_\sigma(G_2 \times J)$.*

Proof. Let $\mathcal{M}$ be any σ-semi-induced matching in $(G \vee H) \times J$. Let $\mathcal{M}_1 = \mathcal{M} \cap E(G_1 \times J)$ and $\mathcal{M}_2 = \mathcal{M} \cap E(G_2 \times J)$. It is clear that $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$, and $\mathcal{M}_1$ and $\mathcal{M}_2$ are σ-semi-induced matchings.

□

Next, we write G_1 and G_2 as $G_1 = G \times_e K_H$ and $G_2 = K_G \times_e H$ as in Lemma 4.2. So, we have that $\mathsf{sim}_\sigma(G_1 \times J) = \mathsf{sim}_{\sigma'}((K_H \times_e G) \times J)$, for some σ', and that $\mathsf{sim}_\sigma(G_2 \times J) = \mathsf{sim}_\sigma((K_G \times_e H) \times J)$ (we can use σ in the second equality since $G_2 \times J = K_G \times_e H$, but we need a different mapping σ' in the first equality since $G_1 \times J$ is only isomorphic to $K_H \times_e G$). Then, by applying associativity in Lemma 4.3, we have that

$$\mathsf{sim}_\sigma((G \vee H) \times J) \leq \mathsf{sim}_{\sigma'}(K_H \times_e (G \times J)) + \mathsf{sim}_\sigma(K_G \times_e (H \times J)). \tag{17}$$

The following lemma will finish the proof.

Lemma 4.7. *For any graph X, set L, and a total order τ on $V(K_L \times X)$, there exists a total order τ' on $V(X)$ such that $\mathsf{sim}_\tau(K_L \times_e X) \leq \mathsf{sim}_{\tau'}(X)$.*

Proof. Let $\mathcal{M}$ be any τ-semi-induced matching in $K_L \times_e X$. We construct a set of edges $\mathcal{M}_X \subseteq E(X)$ by adding to $\mathcal{M}_X$ an edge xy for each edge $(i,x)(j,y) \in E(K_L \times_e X)$. To prove the lemma, it suffices to define a total order τ' on vertices $V(X)$ such that $\mathcal{M}_X$ is a τ'-semi-induced matching in the graph X. We will use Eq. (16) and Eq. (15); we recall them here:

$$xy \in \mathcal{M}_X \implies \exists i,j : (i,x)(j,y) \in \mathcal{M} \tag{15}$$
$$xy \in E(X) \implies \forall i,j : (i,x)(j,y) \in E(K_L \times_e X) \tag{16}$$

Before defining τ', we first argue that $\mathcal{M}_X$ is a matching in X. Suppose otherwise; i.e., $xy, xy' \in \mathcal{M}_X$ for some x, y and y'. Then, from Eq.(15), we have edges $(i,x)(j,y)$ and $(i',x)(j',y')$ in $\mathcal{M}$ for some $i, j, i', j' \in V(K_L)$. This means that $xy, xy' \in E(G)$, and therefore, from Eq.(16), we must also have edges $(i,x)(j',y')$ and $(i',x)(j,y)$. This contradicts the fact that $\mathcal{M}$ is τ-semi-induced matching, i.e. no matter how we define τ, this case cannot happen.

Now, we are ready to define the total order τ' on vertices $V(X)$. Since each vertex in X appears at most once in $\mathcal{M}_X$, for each vertex $x \in V(X)$ that appears in $\mathcal{M}_X$, we define $\tau'(x) = \tau(i,x)$ where (i,x) is the vertex that appears in $\mathcal{M}$. Now, it is easy to check that $\mathcal{M}_X$ is τ'-semi-induced matching.

□

From this lemma, we conclude that there exists a total order σ_1 on $V(K_H \times_e (G \times J))$ such that $\mathsf{sim}_{\sigma'}(K_H \times_e (G \times J)) \leq \mathsf{sim}_{\sigma_1}(G \times J)$, and there exists a total order σ_2 on $V(K_G \times_e (H \times J))$ such that $\mathsf{sim}_{\sigma}(K_G \times_e (H \times J)) \leq \mathsf{sim}_{\sigma_2}(H \times J)$. Theorem 4.5 then follows by combining these inequalities with Eq.(17).

4.3 Subadditivity of Poset Dimension Number (Eq.(3)).

In this section, we show that $\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}(G \times_e \vec{P}) + \chi(G)\mathsf{dim}(H \times_e \vec{P}) + \mathsf{dim}(\vec{P})$. Definitions related to poset and dimension can be found in Section 2. We first note that the extended tensor product between an undirected graph G and a height-two poset $\vec{P}$ is still a poset (in fact, it is a height-two poset). So, the quantity $\mathsf{dim}(G \times_e \vec{P})$ is well-defined. This fact is formalized and proved in the following lemma.

Lemma 4.8. *For any graph A and height-two poset $\vec{P}$, $A \times_e \vec{P}$ is a height-two poset.*

Proof. Consider any vertex $(a,p) \in V(A) \times V(\vec{P})$. Observe that if p is a minimal element in $\vec{P}$, then (a,p) is also a minimal element in $A \times_e \vec{P}$; otherwise, if there is a vertex $(a',p') \in V(A) \times V(\vec{P})$ such that $(a',p')(a,p) \in E(A \times_e \vec{P})$, then $p'p \in E(\vec{P})$, which contradicts the fact that p is minimal in $\vec{P}$. A similar argument shows that if p is a maximal element in $\vec{P}$, then (a,p) is also maximal in $A \times_e \vec{P}$. Since *every* vertex $(a,p) \in V(A) \times V(\vec{P})$ is either a minimal or maximal element (or both) in $A \times_e \vec{P}$, the graph product $A \times_e \vec{P}$ is a height-two poset.

□

Our proof of Eq.(3) has two steps. In the first step (Lemma 4.9), we write the poset $(G \cdot H) \times_e \vec{P}$ as the intersection of two other posets $\vec{P}_1$ and $\vec{P}_2$, where $\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}(\vec{P}_1) + \mathsf{dim}(\vec{P}_2)$. In the second step, we bound the dimensions of $\vec{P}_1$ and $\vec{P}_2$.

Step 1: Decomposition of poset. This step is summarized in the following lemma.

Lemma 4.9. *Consider any undirected graph A and a height-two poset $\vec{P}$. Denote by U and V the set of minimal and maximal elements of $\vec{P}$, respectively. (Since $\vec{P}$ is of height two, $U \cup V = V(\vec{P})$.) Then, $E(A \times_e \vec{P})$ can be written as*

$$E(A \times_e \vec{P}) = E(A \times_e \vec{K}_{U,V}) \cap E(K_A \times_e \vec{P})$$

where $\vec{K}_{U,V}$ is a complete height-two poset with U and V as the sets of minimal and maximal elements respectively, i.e., $E(\vec{K}_{U,V}) = \{uv : u \in U, v \in V\}$.

Proof. The lemma follows from simple logical implications as shown in Fig. 8. The second equality is because $pp' \in E(\vec{P})$ implies that $p \in U$ and $p' \in V$ and the fact that the statement "$a \neq a'$ or $a = a'$" is a true statement.

□

$$
\begin{aligned}
E(A \times_e \vec{P}) &= \{(a,p)(a',p') : (aa' \in E(A) \text{ or } a = a') \text{ and } pp' \in E(\vec{P})\} \\
&= \{(a,p)(a',p') : (aa' \in E(A) \text{ or } a = a') \text{ and } p \in U \text{ and } p' \in V \\
&\quad \text{ and } (a \neq a' \text{ or } a = a') \text{ and } pp' \in E(\vec{P})\} \\
&= \{(a,p)(a',p') : (aa' \in E(A) \text{ or } a = a') \text{ and } p \in U \text{ and } p' \in V\} \cap \\
&\quad \{(a,p)(a',p') : (a \neq a' \text{ or } a = a') \text{ and } pp' \in E(\vec{P})\} \\
&= E(A \times_e \vec{K}_{U,V}) \cap E(K_A \times_e \vec{P})\,.
\end{aligned}
$$

Figure 8: Decomposition of Poset.

Let U and V be as in the above lemma. This allows us to write

$$E((G \cdot H) \times_e \vec{P}) = E((G \cdot H) \times_e \vec{K}_{U,V}) \cap E(K_{G \cdot H} \times_e \vec{P})\,. \tag{18}$$

Note that both $(G \cdot H) \times_e \vec{K}_{U,V}$ and $K_{G \cdot H} \times_e \vec{P}$ are height-two posets (by Lemma 4.8). Moreover, they have the same vertex set, which is $V(G) \times V(H) \times V(\vec{P})$. We next apply the following lemma which relates graph intersection to poset dimension.

Lemma 4.10. *Let $\vec{P}_1$ and $\vec{P}_2$ be any height-two posets on the same vertex set V'. Let $\vec{P} = (V', E(\vec{P}_1) \cap E(\vec{P}_2))$. Then, $\vec{P}$ is a height-two poset. Moreover, $\mathsf{dim}(\vec{P}) \leq \mathsf{dim}(\vec{P}_1) + \mathsf{dim}(\vec{P}_2)$.*

Proof. The proof that $\vec{P}$ is a height-two poset is essentially the same as the proof of Lemma 4.8. Consider any vertex $p \in V'$. Observe that if p is a minimal element in $\vec{P}_1$, then it is also a minimal element in $\vec{P}$; otherwise, if there is a vertex $p' \in V'$ such that $p'p \in E(\vec{P})$, then $p'p \in E(\vec{P}_1)$, which contradicts the fact that p is minimal in $\vec{P}_1$. A similar argument shows that if p is a maximal element in $\vec{P}_1$, then it is also maximal in $\vec{P}$. Since *every* vertex in $\vec{P}$ is either minimal or maximal (or both), $\vec{P}$ is a height-two poset.

We now argue that $\mathsf{dim}(\vec{P}) \leq \mathsf{dim}(\vec{P}_1) + \mathsf{dim}(\vec{P}_2)$. Let $d_i = \mathsf{dim}(\vec{P}_i)$ and let $\varphi_i : V' \to \mathbb{R}^{d_i}$ be a mapping that realizes poset $\vec{P}_i$. We define $\varphi' : V' \to \mathbb{R}^{d_1+d_2}$ as a concatenation of φ_1 and φ_2. That is, for any $p \in V'$, we let $\varphi'(p) = (\varphi_1(p), \varphi_2(p))$. We finish the proof by showing that φ' realizes $\vec{P}$ using the following simple logical implications.

$$
\begin{aligned}
pp' \in E(\vec{P}_1) \cap E(\vec{P}_2) &\iff pp' \in E(\vec{P}_1) \text{ and } pp' \in E(\vec{P}_2) \\
&\iff \varphi_1(p) < \varphi_1(p') \text{ and } \varphi_2(p) < \varphi_2(p') \\
&\iff \varphi'(p) < \varphi'(p')\,.
\end{aligned}
$$

□

Using the above lemma and Eq.(18), we get

$$\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}((G \cdot H) \times_e \vec{K}_{U,V}) + \mathsf{dim}(K_{G \cdot H} \times_e \vec{P}). \tag{19}$$

Step 2: Bounding the dimensions. Our next step is to bound the dimension numbers of $(G \cdot H) \times_e \vec{K}_{U,V}$ and $K_{G \cdot H} \times_e \vec{P}$ separately. The nice thing is that these graph products are not in the general form anymore – one of the graphs in each product is "complete". This allows us to bound the dimension numbers of these graphs as in the next two lemmas.

Lemma 4.11. *For any set L of vertices and any height-two poset $\vec{P}$, $\mathsf{dim}(K_L \times_e \vec{P}) \leq \mathsf{dim}(\vec{P})$.*

Proof. Let $d = \mathsf{dim}(\vec{P})$ and $\varphi : V(\vec{P}) \to \mathbb{R}^d$ be a mapping that realizes $\vec{P}$. We define $\varphi' : V(K_L) \times V(\vec{P}) \to \mathbb{R}^d$ as $\varphi'(v,p) = \varphi(p)$. We complete the proof with the fact that φ' realizes $K_L \times \vec{P}$, proved as follows.

$$\begin{aligned}
&(v,p)(v',p') \in E(K_L \times_e \vec{P}) \\
&\iff ((vv' \in E(K_L)) \text{ or } v = v') \text{ and } pp' \in E(\vec{P}) \\
&\iff pp' \in E(\vec{P}) \\
&\iff \varphi(p) < \varphi(p') \\
&\iff \varphi'(v,p) < \varphi'(v',p')\,.
\end{aligned}$$

□

We note that the equality can be attained in Lemma 4.11, but it is not important to us. The same holds for the next lemma.

Lemma 4.12. *For any undirected graph A and sets U and V of vertices, $\mathsf{dim}(A \times_e \vec{K}_{U,V}) \leq \mathsf{dim}(A \times_e \vec{K}_2)$, where $\vec{K}_{U,V}$ is as in Lemma 4.9.*

Proof. Recall that the set of minimal and maximal elements of $\vec{K}_{U,V}$ are U and V, respectively. Let the minimal and maximal element of $\vec{K}_2$ be u and v, respectively. Let $d = \mathsf{dim}(A \times_e \vec{K}_2)$ and $\varphi : V(A) \times V(\vec{K}_2) \to \mathbb{R}^d$ be a mapping that realizes $A \times_e \vec{K}_2$. Define a function $r : V(\vec{K}_{U,V}) \to V(\vec{K}_2)$ as $r(i) = u$ for all $i \in U$ and $r(i) = v$ otherwise. Now, define a mapping $\varphi' : V(A) \times V(\vec{K}_{U,V}) \to \mathbb{R}^d$ as

$$\varphi'(a,i) = \varphi(a, r(i)).$$

We finish the lemma by showing that φ' realizes poset $A \times_e \vec{K}_{U,V}$. Observe that for any i and i' in $V(\vec{K}_{U,V})$, we have $ii' \in E(\vec{K}_{U,V}) \iff r(i)r(i') \in E(\vec{K}_2)$. Thus,

$$\begin{aligned}
&(a,i)(a',i') \in E(A \times_e \vec{K}_{U,V}) \\
&\iff (aa' \in E(A) \text{ or } a = a') \text{ and } ii' \in E(\vec{K}_{U,V}) \\
&\iff (aa' \in E(A) \text{ or } a = a') \text{ and } r(i)r(i') \in E(\vec{K}_2) \\
&\iff (a, r(i))(a', r(i')) \in E(A \times_e \vec{K}_2) \\
&\iff \varphi(a, r(i)) < \varphi(a', r(i')) \\
&\iff \varphi'(a,i) < \varphi'(a',i')
\end{aligned}$$

□

Applying Lemma 4.11 and 4.12 to Eq.(19), we get

$$\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}((G \cdot H) \times_e \vec{K}_2) + \mathsf{dim}(\vec{P})\,.$$

Finally, we apply Eq.(6) proved in Section 3.3 to get the desired inequality:

$$\mathsf{dim}((G \cdot H) \times_e \vec{P}) \leq \mathsf{dim}(G \times_e \vec{P}) + \chi(G)\mathsf{dim}(H \times_e \vec{P}) + \mathsf{dim}(\vec{P}).$$

5 Hardness from Graph Products

In this section, we show applications of subadditivity inequalities presented in Theorem 1.1 in proving the tight hardness of approximating the induced matching number, the semi-induced matching number, and the poset dimension number. Moreover, we prove the hardness of $d^{1/2-\epsilon}$ for approximating the induced matching number of d-regular bipartite graphs.

5.1 Bipartite Induced Matching.

In this section, we prove that induced and semi-induced matching problems in bipartite graphs are hard to approximate to within a factor of $n^{1-\epsilon}$. Recall that we use $\mathrm{B}[G] = G \times K_2$ and $\mathrm{B}_e[G] = G \times_e K_2$.

We have already sketched the proof of the hardness of the induced matching problem in Section 1.1 and will give more detail here. We can actually say something stronger than just the hardness of these problems. In fact, it is hard to distinguish between the case where an input graph G has large $\mathsf{im}(G)$ and small $\mathsf{sim}(G)$, as stated in Theorem 5.1 below.

Theorem 5.1. *Given any bipartite graph G and $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{ZPP}$, no polynomial-time algorithm can distinguish between the following two cases:*

- *(Yes-Instance) $\mathsf{im}(G) \geq |V(G)|^{1-\epsilon}$.*
- *(No-Instance) $\mathsf{sim}(G) \leq |V(G)|^{\epsilon}$.*

Note that $\mathsf{sim}(G) \geq \mathsf{im}(G)$; thus, Theorem 5.1 implies that no polynomial-time algorithm can distinguish between the cases where $\mathsf{im}(G) \geq |V(G)|^{1-\epsilon}$ and $\mathsf{im}(G) \leq |V(G)|^{\epsilon}$ as well as the cases where $\mathsf{sim}(G) \geq |V(G)|^{1-\epsilon}$ and $\mathsf{sim}(G) \leq |V(G)|^{\epsilon}$. Theorem 5.1 thus implies the hardness of both induced and semi-induced matching problems in bipartite graphs.

Proof of Theorem 5.1. Our proof is based on a reduction from the maximum independent set problem. As discussed earlier, we start from the result of [28] instead of [33], to keep the parameters simple.

Theorem 5.2 ([28])**.** *Let $\epsilon > 0$ be any constant. Given a graph G, unless $\mathsf{NP}=\mathsf{ZPP}$, no polynomial-time algorithm can distinguish between the following two cases:*

- *(Yes-Instance) $\alpha(G) \geq |V(G)|^{1-\epsilon}$.*
- *(No-Instance) $\alpha(G) \leq |V(G)|^{\epsilon}$.*

We start from a graph G given by Theorem 5.2 and return as an output a graph $\mathrm{B}_e[G^k]$, where $k = (1/\epsilon)$ and $G^k = G \vee G \vee \ldots \vee G$ (there are k copies of G). By construction, the number of vertices in $\mathrm{B}_e[G^k]$ is $n = 2|V(G)|^k$.

If G is a Yes-Instance, then we know that $\alpha(G) \geq |V(G)|^{1-\epsilon}$. We will use the following lemma, essentially due to [17], but since it is not explicitly stated in [17], we shall provide the proof for completeness.

Lemma 5.3 (Implicit in [17])**.** *For any graph G,*

$$\alpha(G) \leq \mathsf{im}(B_e[G])\,.$$

Proof. First, we show the lower bound of $\mathsf{im}(\mathrm{B}_e[G])$. Let $I \subseteq V(G)$ be an independent set in G. Clearly, the set of edges $\mathcal{M} = \{(u,1)(u,2) : u \in I\}$ corresponding to I is an induced matching in $\mathrm{B}[G]$ since if there is $u, u' \in I$ such that $(u,1)(u',2) \in \mathcal{M}$, then $uu' \in E(G)$, contradicting the fact that I is an independent set. Thus, $\mathrm{B}[G]$ has an induced matching of size at least $\alpha(G)$. $\square$

We recall the following standard fact in graph theory.

Lemma 5.4 (Folklore; See e.g. [47])**.** *For any graphs G and H, $\alpha(G \vee H) = \alpha(G)\alpha(H)$. In particular, $\alpha(G^k) = (\alpha(G))^k$.*

It follows from the above lemmas that $\mathsf{im}(\mathrm{B}_e[G^k]) \geq \alpha(G^k) \geq (\alpha(G))^k \geq |V(G)|^{k(1-\epsilon)} = \Omega(n^{1-\epsilon})$.

Now if G is a NO-INSTANCE, we can invoke the next lemma, implicitly used in 5.3.

Lemma 5.5. *For any graph G,we have*

$$\mathsf{sim}(B_e[G]) \leq \mathsf{sim}(B[G]) + \alpha(G)\,.$$

Proof. Let σ be any total order on $V(G)$. Consider any σ-semi-induced matching $\mathcal{M}$ in $\mathrm{B}_e[G]$. We may write $\mathcal{M} = \mathcal{M}_1 \cup \mathcal{M}_2$, where $\mathcal{M}_1$ consists of edges that are also present in $\mathrm{B}[G]$ and $\mathcal{M}_2 = \mathcal{M} \setminus \mathcal{M}_1$. It can be seen that $|\mathcal{M}_1| \leq \mathsf{sim}_\sigma(\mathrm{B}[G])$ and $|\mathcal{M}_2| \leq \alpha(G)$. The former inequality is because $E(\mathrm{B}[G]) \subseteq E(\mathrm{B}_e[G])$. The latter inequality is because we can define an independent set in G by choosing vertices corresponding to edges in $\mathcal{M}_2$ (since every edge in $\mathcal{M}_2$ is in the form $(v, 1)(v, 2)$ for some $v \in V(G)$). Since this is true for all σ, the lemma follows.

$\square$

Corollary 5.6 (Immediate from Theorem 1.1)**.** *For any integer k and graph G, $\mathsf{sim}(B[G^k]) \leq \mathsf{sim}(B[G])k$.*

By applying Lemma 5.5, we have that $\mathsf{sim}(\mathrm{B}_e[G^k]) \leq \mathsf{sim}(\mathrm{B}[G^k]) + \alpha(G)^k$, and by invoking Corollary 5.6, we have $\mathsf{sim}(\mathrm{B}_e[G^k]) \leq \mathsf{sim}(\mathrm{B}[G])k + \alpha(G)^k$. Then we plug in $k = 1/\epsilon$ and $\alpha(G) \leq |V(G)|^\epsilon$ and conclude that $\mathsf{sim}(\mathrm{B}_e[G^k]) \leq O(|V(G)|) \leq n^{2\epsilon}$ when G is a NO-INSTANCE. This completes the proof of Theorem 5.1.

To get a better hardness result, we start the reduction from Theorem 1 in [33] using the value of $k = \log^\gamma |V(G)|$ instead of $(1/\epsilon)$ (where γ is as in Theorem 1 in [33]). This will give the hardness of $n/2^{\log^{3/4+\gamma} n}$ under the assumption that $\mathsf{NP} \not\subseteq \mathsf{BPTIME}(n^{\mathrm{poly}\log n})$.

5.2 Induced and Semi-induced Matchings on d-Regular Graphs.

Here we show the hardness result of the induced matching problem on d-regular bipartite graphs. For this, we need an instance G of the maximum independent set problem such that G is d-regular. It can be seen that the following hardness result follows from Trevisan's construction in [46] on the hardness of the maximum independent set problem on bounded degree graphs. As it is not guaranteed that an instance G obtained from Trevisan's construction has regular degree, we have to slightly modify the construction in the same way as in [13] and [5].

Theorem 5.7 ([46], modified from Theorem 4 in [13])**.** *Let $\lambda : \mathbb{N} \to \mathbb{N}$ be any function. Assuming that $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\lambda(n))})$, there is no polynomial-time algorithm that can solve the following problem.*

For any constant $\epsilon > 0$ and any integer q, given a graph G of size $q^{O(\lambda(q))}$ such that all vertices have degree $\Delta = 2^{O(\lambda(q))}$, the goal is to distinguish between the following two cases:

- *(*YES-INSTANCE*) $\alpha(G) \geq |V(G)|/\Delta^\epsilon$.*
- *(*NO-INSTANCE*) $\alpha(G) \leq |V(G)|/\Delta^{1-\epsilon}$.*

This theorem gives the hardness result of the bounded degree version of the maximum independent set problem, and it allows us to use the value of λ to specify the degree of vertices we want. For instance, if we use $\lambda(q) = c$ for some constant c, then we get the hardness of the constant-degree maximum independent set problem, and the hardness assumption is $\mathsf{NP} \not\subseteq \mathsf{ZPP}$. But, if we choose $\lambda(q) = O(\log\log q)$, then we have the hardness for the logarithmic-degree maximum independent set problem with the hardness assumption of $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\log\log n)})$.

We will need the following lemma which will also be used later to prove the hardness of pricing problems.

Lemma 5.8. *Let $\Delta : \mathbb{N} \to \mathbb{N}$. Assuming that $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\log \Delta(n))})$, there is no polynomial time algorithm that can solve the following problem: For any ϵ and integer q, given a $\Delta(q)$-regular graph G on $O(q^{O(\log \Delta(q))})$ vertices and an empty graph H on $\Delta(q)$ vertices, the goal is to distinguish between the following cases:*

- *(Yes-Instance) $\mathsf{im}(B_e[G \vee H]) \geq |V(G)|(\Delta(q))^{1-\epsilon}$*
- *(No-Instance) $\mathsf{sim}(B_e[G \vee H]) \leq |V(G)|(\Delta(q))^{\epsilon}$.*

Note that we use $\Delta(q) = c$ for some constant c in the proof of Theorem 5.9, while we choose $\Delta(q) = \mathrm{poly}\log q$ when proving the hardness of pricing problems in the next section. From the lemma, the hardness of the induced matching problem on d-regular graphs follows immediately.

Of Lemma 5.8. Let $\epsilon > 0$ be a constant. Let G be a $\Delta(q)$-regular graph obtained from Theorem 5.7 (choosing $\lambda(q) = O(\log \Delta(q))$ so that we get the graph G of degree $\Delta(q)$ and $|V(G)| = q^{O(\log \Delta(q))}$.) We output a graph $\mathrm{B}_e[G \vee H]$, where $H = \bar{K}_{\Delta(q)}$ is an empty graph on $\Delta(q)$ vertices. This finishes the construction.

Notice that the number of vertices in $\mathrm{B}_e[G \vee H]$ is $n = 2|V(G)||V(H)| = 2|V(G)|\Delta(q)$. In the Yes-Instance, by Lemma 5.3 and 5.4, we have that $\mathsf{im}(\mathrm{B}_e[G \vee H]) \geq \alpha(G \vee H) \geq \alpha(G)\alpha(H) \geq |V(G)|\Delta(q)^{1-\epsilon}$ because $\alpha(G) \geq |V(G)|/\Delta(q)^{\epsilon}$ in the Yes-Instance.

In the No-Instance, we have $\mathsf{sim}(\mathrm{B}_e[G \vee H]) \leq \alpha(G \vee H) + \mathsf{sim}(\mathrm{B}[G \vee H])$ (by Lemma 5.5). The first term is at most $\alpha(G)\alpha(H) \leq |V(G)|\Delta(q)^{\epsilon}$ (by Lemma 5.4). The second term is at most, by Eq.(2) in Theorem 1.1, $\mathsf{sim}(\mathrm{B}[G]) + \mathsf{sim}(\mathrm{B}[H]) \leq |V(G)| + \Delta(q) \leq 2|V(G)|$, for our choice of $\Delta(q)$. Therefore, in the No-Instance, the value of the solution is at most $O(|V(G)|\Delta(q)^{\epsilon})$.

Since $\lambda(q) = O(\log \Delta(q))$, we get the complexity assumption of $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\log \Delta(n))})$. □

Theorem 5.9. *Let $d \leq \mathrm{poly}\log n$ be any sufficiently large number. For any constant $\epsilon > 0$, unless $\mathsf{NP} \subseteq \mathsf{ZPTIME}(n^{\mathrm{poly}\log n})$, it is hard to approximate the induced matching problem on d-regular graphs to within a factor of $d^{1/2-\epsilon}$.*

Proof. From Lemma 5.8, observe that the degree of each vertex in $\mathrm{B}_e[G \vee H]$ is $d = \Delta^2 + 1$: each vertex $(v, a, 1) \in \mathrm{B}_e[G \vee H]$ is connected to $(v, a, 2)$ and other vertices $(u, b, 2)$ for all $u \in V(G)$ and $b \in V(H)$. The gap between Yes-Instance and No-Instance is $\Delta^{1-2\epsilon} \geq d^{1/2-O(\epsilon)}$. We use $\Delta \leq O(\mathrm{poly}\log n)$, so the running time of the reduction is $n^{O(\log\log n)}$. □

5.3 Poset Dimension.

We now prove the $n^{1-\epsilon}$-hardness of approximating poset dimension. Note that here we use $\mathrm{B}[G] = G \times \vec{K}_2$ and $\mathrm{B}_e[G] = G \times_e \vec{K}_2$. We denote by $G^k = G \cdot G \cdot \ldots \cdot G$ where G appears k times.

Construction. We will need the following hardness result of the graph coloring problem, due to Feige and Kilian [19]. (In fact, there is a stronger hardness result by Khot and Ponnuswami [33], but we use the result of Feige and Kilian to keep the presentation simple.)

Theorem 5.10 ([19]). *Let $\epsilon > 0$ be any constant. Given a graph G, unless* $\mathsf{NP}=\mathsf{ZPP}$*, no polynomial-time algorithm can distinguish between the following two cases:*

- *(*YES-INSTANCE*) $\chi(G) \leq |V(G)|^{\epsilon}$.*
- *(*NO-INSTANCE*) $\chi(G) \geq |V(G)|^{1-\epsilon}$.*

Our reduction starts from the instance G given by Theorem 5.10. Then we output $\mathrm{B}[G^k]$ where $k = 1/\epsilon$. The construction size is $n = 2|V(G)|^k$.

Analysis. We need the following lemma, similar in spirit to Lemma 5.3. Since we state the lemma in our language, we provide the proof for completeness.

Lemma 5.11 (Implicit in [29]). *For any graph G, $\chi(G) \leq \mathsf{dim}(B[G]) \leq \mathsf{dim}(B_e[G]) + \chi(G)$*

Proof. Recall that $\mathrm{B}[G]$ is almost identical to $\mathrm{B}_e[G]$ except that $\mathrm{B}[G]$ has no edges of the form $(u,1)(u,2)$ for all $u \in V(G)$. We say that a mapping $\psi : \mathrm{B}_e[G] \to \mathbb{R}$ **hits** a vertex $u \in V(G)$ if $\psi((u,1)) > \psi((u,2))$, and $\psi((v,1)) \leq \psi((w,2))$ whenever $vw \in E(G)$. In other words, ψ is a linear order that "reverses" the direction of edge $(u,1)(u,2) \in \mathrm{B}_e[G]$.

The following claim was proved by Hegde and Jain in [29]. We restated it here in our terminology and also provide the proof for completeness.

Claim 5.12 ([29]). *Let $X \subseteq V(G)$. There is a mapping ψ that hits all vertices in X if and only if X is an independent set in G.*

Proof. One direction is easy to see. Suppose $X \subseteq V(G)$ contains u, v such that $uv \in E(G)$, so the function ψ hitting $\{u, v\}$ means that $\psi((u,2)) > \psi((u,1)) \geq \psi((v,2)) > \psi((v,1)) \geq \psi((u,2))$, an obvious contradiction. In short, we cannot expect to have a linear extension of a directed cycle.

Conversely, assume that X is an independent set. We define function ψ by processing vertices in X in arbitrary order. When vertex $u \in X$ is considered, we set the values $\psi((u,1)) = 2$ and $\psi((u,2)) = 1$. After we finish, we set $\psi((u,1)) = 0$ and $\psi((u,2)) = 3$ for all other vertices u's whose values ψ were undefined. Now notice that the only way to violate the hitting property of ψ is to have $\psi((u,1)) = 2$ and $\psi((v,2)) = 1$ for some $uv \in E(G)$, but this is impossible because X is an independent set.

□

Now we prove the inequality.

The lower bound follows immediately from Claim 5.12. Let $\tilde{\varphi} : V(\mathrm{B}[G]) \to \mathbb{R}^d$ be a mapping that realizes $\mathrm{B}[G]$. For each coordinate q, define ψ_q as $\psi_q((u,1)) = \tilde{\varphi}(((u,1))[q]$ and $\psi_q((u,2)) = \tilde{\varphi}((u,2))[q]$, i.e. ψ_q is function where we project the qth coordinate of $\tilde{\varphi}$. Observe that, for each vertex $u \in V(G)$, there must be some ψ_q that hits the vertex u. We argue that there is a valid coloring of G using at most d colors. To see this, construct a coloring as follows. For each vertex $u \in V(G)$, assign a color q to u, where q is the first coordinate such that ψ_q hits the vertex u. Claim 5.12 guarantees that each color class is an independent set. Thus, the coloring is valid and $\chi(G) \leq d$.

To prove the upper bound, let φ be a function that realizes the poset $\mathrm{B}_e[G]$. Then, for any vertices $u \neq v$ of G, we have $\varphi((u,1)) \leq \varphi((v,2))$ if and only if $uv \in E(G)$. Then we need to extend φ into $\tilde{\varphi}$ such that (i) Each vertex $u \in V(G)$ is hit by some coordinate of $\tilde{\varphi}$, and (ii) For any two vertices of the form (u,i) and (v,i) where $u \neq v$ and $i \in \{1,2\}$, we must have some coordinates q, q' such that $\tilde{\varphi}((u,1))[q] < \tilde{\varphi}((v,1))[q]$ and $\tilde{\varphi}((u,1))[q'] > \tilde{\varphi}((v,1))[q']$. We only need two more coordinates to satisfy (ii). To deal with (i), it suffices to find a collection of functions ψ_j such that each vertex $u \in V(G)$ is hit by some ψ_j. Then $\tilde{\varphi}$ can be defined by concatenating φ with all the mappings ψ_j. Each such ψ_j can be obtained from Claim 5.12 by defining, for each j, ψ_j to be a map that hits all vertices in a color class j.

$\square$

We will also need the following lemma which bounds the chromatic number of the k-fold product of graphs.

Lemma 5.13 ([37, 22, 34] and [42, Cor. 3.4.5]))**.** *For any graph G and any number k, $\left(\frac{\chi(G)}{\log|V(G)|}\right)^k \leq \chi(G^k) \leq (\chi(G))^k$.*

We are now ready to analyze the gap between the Yes-Instance and No-Instance.

Suppose that G is a No-Instance. Then $\chi(G) \geq |V(G)|^{1-\epsilon}$. By Lemma 5.11 and 5.13 and for sufficiently large $|V(G)|$ (so that $\log|V(G)| \leq |V(G)|^{\epsilon}$), we have that $\mathsf{dim}(\mathrm{B}[G^k]) \geq \chi(G^k) \geq \left(\frac{\chi(G)}{\log|V(G)|}\right)^k \geq \left(\frac{|V(G)|^{1-\epsilon}}{\log|V(G)|}\right)^k \geq \frac{n^{1-\epsilon}}{(2\log|V(G)|)^k} \geq n^{1-O(\epsilon)}$.

For the Yes-Instance, we have that $\mathsf{dim}(\mathrm{B}[G^k]) \leq \mathsf{dim}(\mathrm{B}_e[G^k]) + \chi(G^k)$. By Lemma 5.13, the term $\chi(G^k)$ can be upper bounded by $\chi(G)^k \leq |V(G)|^{\epsilon k} = |V(G)| \leq n^{\epsilon}$ because $\chi(G) \leq |V(G)|^{\epsilon}$ in the Yes-Instance. We use the following claim to bound the term $\mathsf{dim}(\mathrm{B}_e[G^k])$.

Claim 5.14. *For any graph G and integer k, $\mathsf{dim}(B_e[G^k]) \leq \chi(G)\mathsf{dim}(B_e[G])k + k$.*

Proof. Recall that $\mathrm{B}[G]$ is almost identical to $\mathrm{B}_e[G]$ except that $\mathrm{B}[G]$ has no edges of the form $(u,1)(u,2)$ for all $u \in V(G)$. We say that a mapping $\psi : \mathrm{B}_e[G] \to \mathbb{R}$ **hits** a vertex $u \in V(G)$ if $\psi((u,1)) > \psi((u,2))$, and $\psi((v,1)) \leq \psi((w,2))$ whenever $vw \in E(G)$. In other words, ψ is a linear order that "reverses" the direction of edge $(u,1)(u,2) \in \mathrm{B}_e[G]$.

Now, we prove the inequality.

The lower bound follows immediately from Claim 5.12. Let $\tilde{\varphi} : V(\mathrm{B}[G]) \to \mathbb{R}^d$ be a mapping that realizes $\mathrm{B}[G]$. For each coordinate q, define ψ_q as $\psi_q((u,1)) = \tilde{\varphi}(((u,1))[q]$ and $\psi_q((u,2)) = \tilde{\varphi}((u,2))[q]$, i.e. ψ_q is function where we project the qth coordinate of $\tilde{\varphi}$. Observe that, for each vertex $u \in V(G)$, there must be some ψ_q that hits the vertex u. We argue that there is a valid coloring of G using at most d colors. To see this, construct a coloring as follows. For each vertex $u \in V(G)$, assign a color q to u, where q is the first coordinate such that ψ_q hits the vertex u. Claim 5.12 guarantees that each color class is an independent set. Thus, the coloring is valid and $\chi(G) \leq d$.

To prove the upper bound, let φ be a function that realizes the poset $\mathrm{B}_e[G]$. Then, for any vertices $u \neq v$ of G, we have $\varphi((u,1)) \leq \varphi((v,2))$ if and only if $uv \in E(G)$. Then we need to extend φ into $\tilde{\varphi}$ such that (i) Each vertex $u \in V(G)$ is hit by some coordinate of $\tilde{\varphi}$, and (ii) For any two vertices of the form (u,i) and (v,i) where $u \neq v$ and $i \in \{1,2\}$, we must have some coordinates q, q' such that $\tilde{\varphi}((u,1))[q] < \tilde{\varphi}((v,1))[q]$ and $\tilde{\varphi}((u,1))[q'] > \tilde{\varphi}((v,1))[q']$. We only need two more coordinates to satisfy (ii). To deal with (i), it suffices to find a collection of functions ψ_j such that each vertex $u \in V(G)$ is hit by some ψ_j. Then $\tilde{\varphi}$ can be defined by concatenating φ with all the mappings ψ_j. Each such ψ_j can be obtained from Claim 5.12 by defining, for each j, ψ_j to be a map that hits all vertices in a color class j.

$\square$

Using the claim, we have that $\mathsf{dim}(\mathrm{B}_e[G^k]) \leq \chi(G)\mathsf{dim}(\mathrm{B}_e[G])k + k \leq 2|V(G)|^{\epsilon}|V(G)|k + k$, since $\mathsf{dim}(\mathrm{B}_e[G]) \leq 2|V(G)|$. This implies that $\mathsf{dim}(\mathrm{B}_e[G^k]) \leq O(|V(G)|^{1+\epsilon}) \leq n^{2\epsilon}$ and therefore, $\mathsf{dim}(\mathrm{B}[G^k]) \leq n^{O(\epsilon)}$. We now have the gap of $n^{1-O(\epsilon)}$.

6 More Applications

In this section, we present all other applications discussed in the introduction.

6.1 Maximum Feasible Subsystem with 0/1 Coefficients.

The following reduction follows the ideas implicit in Theorem 3.5 in [17]. For completeness, we include the proof in Appendix A.

Theorem 6.1 ([17])**.** *Consider an instance $G = (V_1 \cup V_2, E)$ of the bipartite semi-induced matching problem. There is a polynomial time reduction that, for any $0 < \beta \le |V(G)|$, outputs an instance $\mathcal{A} = (A, \ell, \mu)$ of* Mrfs *satisfying the following properties:*

- *(*Size*) Matrix A is an m-by-n matrix, where $m = |V_1|$, $n = |V_2|$ and $L = \max_{i\in[n]} \{\ell_i\} \le (\beta n)^{O(n)}$.*
- *(*Yes-Instance*) There is a solution $\mathbf{x} \in \mathbb{R}^n_+$ that satisfies at least $\mathsf{im}(G)$ constraints in $\mathcal{A}$.*
- *(*No-Instance*) There is no solution $\mathbf{x} \in \mathbb{R}^n_+$ that "β-satisfies" more than $\mathsf{sim}(G)$ constraints in $\mathcal{A}$; i.e., $|\left\{i : \ell_i \le a_i^T\mathbf{x} \le \beta\mu_i\right\}| \le \mathsf{sim}(G)$ for all $\mathbf{x}$.*

Now, we prove the hardness of approximating Mrfs, which holds even in the following *bi-criteria* setting. For any instance $\mathcal{A}$, we denote by $\mathsf{OPT}(\mathcal{A})$ the maximum number of constraints that can be satisfied, i.e., $\mathsf{OPT}(\mathcal{A}) = \max_i |\left\{i : \ell_i \le a_i^T\mathbf{x} \le \mu_i\right\}|$. For any $0 < \alpha, \beta \le m$, we say that an algorithm is an (α, β)-approximation algorithm if, for any instance $\mathcal{A}$ of Mrfs, the algorithm returns a solution $\mathbf{x}$ that β-satisfies at least $\mathsf{OPT}(\mathcal{A})/\alpha$ constraints; i.e., $|\left\{i : \ell_i \le a_i^T\mathbf{x} \le \beta\mu_i\right\}| \ge \mathsf{OPT}(\mathcal{A})/\alpha$. The non-bi-criteria setting (defined in Section 1.2) is when $\beta = 1$.

Corollary 6.2. *Let $\epsilon > 0$ be any constant. There is no polynomial-time $(m^{1-\epsilon}, m+n)$-approximation algorithm for* Mrfs *unless $\mathsf{NP} \subseteq \mathsf{ZPP}$. Moreover, when considering an approximation factor in terms of L, finding $(\log^{1-\epsilon} L, m + n)$-approximation algorithm cannot be done in polynomial time, unless $\mathsf{NP} \subseteq \mathsf{ZPP}$.*

Proof. We start from the graph G given by Theorem 5.1 and invoke Theorem 6.1 on G. For Yes-Instance where $\mathsf{im}(G) \ge |V(G)|^{1-\epsilon}$, we have that there is a solution $\mathbf{x}$ that satisfies $\mathsf{im}(G) \ge |V(G)|^{1-\epsilon} = m^{1-\epsilon}$ constraints. In the No-Instance where $\mathsf{sim}(G) \le |V(G)|^{\epsilon}$, there is no solution that β-satisfies $\mathsf{sim}(G) \le |V(G)|^{\epsilon} \le m^{\epsilon}$, for any $0 < \beta \le |V(G)| = m + n$. Thus, even when we allow the solution to $(m + n)$-satisfies the constraints, there is still an $m^{1-O(\epsilon)}$ gap.

Theorem 6.1 guarantees that $L \le 2^{O(n \log n)} \le 2^{n^{1+\epsilon}}$, so the hardness of $n^{1-O(\epsilon)}$ can also be written as $\log^{1-O(\epsilon)} L$. $\square$

We remark that the hardness factor can be improved to $m/2^{\log^{3/4+\gamma} m}$. Our bounds are nearly tight since it is trivial to get a factor of m-approximation, and since [17] showed an $O(\log(nL))$-approximation algorithm.

6.2 Pricing Problems.

In this section, we revisit Udp-Min and Smp and give an alternative proof of the hardness results in [13]. As discussed in the introduction, our proof illustrates the insight that the maximum expanding sequence problem, which is equivalent to the bipartite semi-induced matching problem (see Appendix A.2), is the main source of hardness of these pricing problems.

We start by defining the pricing problems we consider. In Unit-Demand Min-Buying Pricing (Udp-Min), we have a collection of items $\mathcal{I} = [n]$ and customers $\mathcal{C}$ where each consumer $c \in \mathcal{C}$ is associated with set $S_c \subseteq [n]$ and a budget B_c. Once the price $p : \mathcal{I} \to \mathbb{R}_+$ is fixed, each consumer c buys the cheapest item in S_c if the price of such item is at most B_c; otherwise, the consumer buys nothing. Our goal is to set the price p so that the profit is maximized.

In Single-Minded Pricing (Smp), the setting is the same except that now each consumer c buys the whole set S_c of its items if $\sum_{i \in S_c} p(i) \le B_c$; otherwise, the consumer c buys nothing. Again, the goal is to set the prices p so that the profit is maximized.

For any instance $\mathcal{P}$ of Udp-Min or Smp, we denote by $\mathsf{OPT}(\mathcal{P})$ the revenue that can be obtained by an optimal price function.

Our contribution lies in proving the following theorem that makes connections between the bipartite semi-induced matching problem and pricing problems. The proof of this theorem borrows many ideas from [10, 13]. The proof is included in Appendix A.

Theorem 6.3. *There are reductions with a running time of $|V(G)|^{O(|V(H)|)}$ that transform input graph $G' = B_e[G \vee H]$ into an instance $(\mathcal{C}, \mathcal{I})$ of* Udp-Min *or* Smp *such that*

$$\mathsf{im}(G') \le \mathsf{OPT}(\mathcal{C}, \mathcal{I}) \le 2\mathsf{sim}(G') + O(|V(G)|(1 + |E(H)|))$$

Furthermore, the number of consumers and items are $|\mathcal{C}| = |V(H)|^{O(|V(H)|)}|V(G)|$ and $|\mathcal{I}| = |V(G)||V(H)|$, respectively, and each consumer $c \in \mathcal{C}$ satisfies $|S_c| \le O(\Delta^2)$.

Note that the running time and the number of consumers for Udp-Min can be slightly improved with a more careful analysis as follows: The running time can be made $2^{O(|V(H)|)}\,\mathrm{poly}\,|V(G)|$, and the number of consumers is $2^{O(|V(H)|)}|V(G)|$.

Applying the subadditivity property (Theorems 1.1), the hardness of the induced and semi-induced matching problems (Lemma 5.8) and Theorem 6.3, we get the following result, which is an alternative proof of the result in [13].

Theorem 6.4. *For any constant $\epsilon > 0$, both* Smp *and* Udp-Min *are hard to approximate to within a factor of $\log^{1-\epsilon}|\mathcal{C}|$ and $(\max_{c \in \mathcal{C}} |S_c|)^{1/2-\epsilon}$, where $\mathcal{C}$ is the set of consumers, unless $\mathsf{NP} \subseteq \mathsf{ZPTIME}(n^{O(\mathrm{poly}\log n)})$.*

Proof. First, we take a $\Delta(q)$-regular graph G on $O(q^{O(\log \Delta(q))})$ vertices and an empty graph H on $\Delta(q)$ vertices as stated in Lemma 5.8. Thus, assuming that $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\log \Delta(n))})$, there is no polynomial-time algorithm that distinguishes between the case that $\mathsf{im}(\mathrm{B}_e[G \vee H]) \ge |V(G)|(\Delta(q))^{1-\epsilon}$ and the case that $\mathsf{sim}(\mathrm{B}_e[G \vee H]) \le |V(G)|(\Delta(q))^{\epsilon}$, for function Δ whose value will be specified later.

Then we apply Theorem 6.3 on $\mathrm{B}_e[G \vee H]$ to obtain an instance $(\mathcal{C}, \mathcal{I})$ of Udp-Min. This means that in the Yes-Instance, the optimal revenue from $(\mathcal{C}, \mathcal{I})$ is at least $\mathsf{im}(G') \ge |V(G)|(\Delta(q))^{1-\epsilon}$. Additionally, the optimal revenue in the No-Instance is at most $2\mathsf{sim}(G') + O(|V(G)|(1 + |E(H)|))$, which is $O(|V(G)|(\Delta(q))^{\epsilon})$ because $\mathsf{sim}(G') \le |V(G)|(\Delta(q))^{\epsilon}$ (by Lemma 5.8), and the term $|V(G)|(1 + |E(H)|)$ is at most $O(|V(G)|) \le |V(G)|\Delta^{\epsilon}$ since H is an empty graph. This implies the gap of $(\Delta(q))^{1-O(\epsilon)}$ between the two cases.

Now, we choose $\Delta(q) = \log^b q$ where $b = O(1/\epsilon)$. So, $|V(H)| = \Delta(q) = \log^b q$ and $|V(G)| = q^{O(\log\log q)}$. By Theorem 6.3, the number of consumers is bounded by $|\mathcal{C}| \le 2^{O(|V(H)|)}|V(G)| \le 2^{\log^{b+1} q} = 2^{(\Delta(q))^{1+\epsilon}}$, so we obtain the gap of $(\Delta(q))^{1-O(\epsilon)} \ge \log^{1-O(\epsilon)}|\mathcal{C}|$ as desired.

Lemma 5.8 holds with the assumption that $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{O(\log \Delta(n))}) = \mathsf{ZPTIME}(n^{O(\log\log n)})$, and the running time of the reduction in Theorem 6.3 is $|V(G)|^{O(|V(H)|)} = q^{\log^b q} = q^{\mathrm{poly}\log q}$, so the hardness assumption we need is $\mathsf{NP} \not\subseteq \mathsf{ZPTIME}(n^{\mathrm{poly}\log n})$.

Now to get the hardness in terms of $k^{1/2-\epsilon} = (\max_{c\in\mathcal{C}} |S_c|)^{1/2-\epsilon}$, notice that $k \leq O(\Delta(q)^2)$. Therefore, the hardness in terms of k is $(\Delta(q))^{1-O(\epsilon)} = k^{1/2-O(\epsilon)}$. This holds for any $\Delta(q) \leq \operatorname{poly}\log q$.

□

6.3 Donation Center Location Problem.

The Donation Center Location (DCL) problem is defined as follows. The input consists of a directed bipartite graph $G = (A \cup L, E)$ with edges directed from the set A of *agents* to the set L of *donation centers*. Each center $\ell \in L$ has a capacity $c_\ell \in \mathbb{Z}$ that represents the maximum number of clients that can be served, and each vertex $a \in A$ has a strictly ordered preference ranking of its neighbor in L. We are interested in choosing a subset $L' \subseteq L$ of centers to open and an assignment of subset $A' \subseteq A$ of agents to centers such that: (1) The number of agents assigned to any center ℓ is at most c_ℓ, and (2) Each $a \in A'$ is assigned to its highest-ranked neighbor in L'. Our goal is to maximize the number of satisfied agents.

We therefore write an instance of DCL as a triple $\mathcal{P} = (G = (A \cup L, E), \{c_\ell\}_{\ell\in L}, \{\preceq_a\}_{a\in A})$, where the relation $\preceq_a$ represents the rank relation of agent $a \in A$. Denote by $\mathsf{OPT}(\mathcal{P})$ the optimal value of the instance $\mathcal{P}$. The following theorem makes a connection between DCL and the semi-induced matching problem.

Theorem 6.5. *Let G' be a bipartite graph. There is a polynomial time reduction that transforms G' into an instance $\mathcal{P} = (G, \{c_\ell\}, \{\preceq_a\})$ of DCL such that:*

$$\mathsf{im}(G') \leq \mathsf{OPT}(\mathcal{P}) \leq \mathsf{sim}(G')$$

Moreover, $|V(G)| = |V(G')|$, $c_\ell = 1$ for all $\ell \in L$, and $\preceq_a = \preceq^*$ for all $a \in A$, where $\preceq^*$ is some global preference.

Proof. Given a bipartite graph $G' = (V_1 \cup V_2, E)$ where $V_1 = \{(u,1) : u \in [n]\}$ and $V_2 = \{(u,2) : u \in [n]\}$, we create an instance of DCL as follows. Each vertex $(u,1)$ represents a center $\ell(u)$, and each vertex $(v,2)$ represents an agent $a(v)$. The capacity of each center $\ell(u)$ is $c_{\ell(u)} = 1$, and each agent uses a global preference $\preceq^*$ that satisfies $\ell(u) \preceq^* \ell(u')$ if and only if $u > u'$ are integers.

First, we prove the lower bound of $\mathsf{OPT}(\mathcal{P}) \geq \mathsf{im}(G')$. Given any induced matching $\mathcal{M}$ in G', we argue that there is a solution of value $|\mathcal{M}|$ to the DCL instance. For each edge $(u,1)(v,2) \in \mathcal{M}$, we open the center $\ell(u)$ and assign the agent $a(v)$ to $\ell(u)$. Now, we only need to argue that all agents that are matched in the matching $\mathcal{M}$ are satisfied. For each $(v,2)$, where $(u,1)(v,2) \in \mathcal{M}$, assume that $(v,2)$ prefers some other center $\ell(u')$ to $\ell(u)$ that is currently open. This means that there must be an edge $(u',1)(v,2) \in E$, contradicting the fact that $\mathcal{M}$ is an induced matching.

To prove the upper bound, given any solution $L' \subseteq L, A' \subseteq A$ and assignment $\varphi : A' \to L'$, we show that we can construct a σ-semi-induced matching $\mathcal{M}$ in G' such that $|\mathcal{M}| = |A'|$. Consider the following set:

$$\mathcal{M} = \{(u,1)(v,2) : \ell(u) \in L' \text{ and } \varphi(a(v)) = \ell(u)\}$$

The set $\mathcal{M}$ is indeed a matching because each agent is only assigned once, and each center has unit capacity. It is sufficient to show that the matching $\mathcal{M}$ is σ-semi-induced for total order σ defined by $\sigma(u) < \sigma(u')$ if and only if $u < u'$. Assume that this is not a σ-semi-induced matching. Then there must be an edge $(u',1)(v,2)$, but $\varphi(a(v)) = \ell(u)$ and $u > u'$. This means that the agent $a(v)$ prefers $\ell(u')$ to $\ell(u)$, but $a(v)$ was assigned to $\ell(u)$ instead. This contradicts the fact that the solution is feasible.

□

Corollary 6.6. *Let $\epsilon > 0$ be any constant. Unless $\mathsf{NP} \subseteq \mathsf{ZPP}$, it is hard to approximate* DCL *to within a factor of $n^{1-\epsilon}$ where n is the number of vertices in the input graph.*

Proof. First, we take a graph G' as in Theorem 5.1, and we then invoke Theorem 6.5 on G' to obtain an instance $\mathcal{P} = (G = (A \cup L, E), \{c_l\}, \{\preceq_a\})$ of DCL. In the YES-INSTANCE, where $\mathsf{im}(G') \geq |V(G')|^{1-\epsilon}$, we have that $\mathsf{OPT}(\mathcal{P}) \geq |V(G)|1 - \epsilon$. In the NO-INSTANCE, where $\mathsf{sim}(G') \leq |V(G')|^{\epsilon}$, we have $\mathsf{OPT}(\mathcal{P}) \leq |V(G)|^{\epsilon}$. Therefore, we obtain a gap of $|V(G)|^{1-2\epsilon}$ as desired. $\square$

6.4 Boxicity, Cubicity, and Threshold Dimension.

We start by giving the definitions of the problems and related notions in graph theory.

Definition 6.7 (Intersection Graph)**.** *Given a graph $G = (V, E)$, we say that a set system $\{S_v\}_{v \in V(G)}$ is a* set system representation *of G if*

$$(\forall u, v \in V(G))uv \in E \textit{ iff } S_u \cap S_v \neq \emptyset$$

It is well-known that any graph G can be represented by a set system: for each vertex $u \in V(G)$, we define set S_u to contain edges incident to u. We are interested in a set system representation where each set corresponds to a geometric object.

Definition 6.8 (Boxicity and Cubicity)**.** *We say that the* boxicity *of a graph G is at most d if G can be represented by a set system $\{S_v\}_{v \in V(G)}$ such that each set S_v is a d-dimensional axis-parallel hyper-rectangle in $\mathbb{R}^d$. Similarly, we say that the* cubicity *of G is at most d if each set S_v is a unit cube in $\mathbb{R}^d$.*

In other words, the boxicity of G, denoted by $\mathsf{box}(G)$, is the minimum dimension d such that we can represent each node $v \in V(G)$ as a d-dimensional rectangle in the geometric setting. It is known that the boxicity is at most one and two in interval graphs and planar graphs, respectively.

Definition 6.9 (Threshold Dimension)**.** *A graph G is a* threshold graph *if there is a real number η and weight function $w : V(G) \to \mathbb{R}$ such that $uv \in E(G) \Leftrightarrow w(u) + w(v) \geq \eta$. For any graph G, the* threshold dimension *of G is the minimum k such that there exist threshold graphs $G_1, \ldots, G_k$ where $E(G) = \bigcup_{i=1}^{k} E(G_i)$.*

Adiga et al. [1] show that the problems of approximating boxicity, cubicity, and threshold dimension are at least as hard as poset dimension (within a constant factor). We get the tight hardness of these problems by combining the reductions in [1] with our hardness of poset dimension. We provide an outline of their reductions here and the readers to [1] for the complete proof.

First, they show that there is a polynomial-time algorithm that transforms any poset $\vec{P}$ into a graph $G_{\vec{P}}$ such that $\mathsf{box}(G_{\vec{P}}) \leq \mathsf{dim}(\vec{P}) \leq 2\mathsf{box}(G_{\vec{P}})$. This implies the hardness of approximating boxicity of graphs. Since cubicity is known to be within a logarithmic factor of boxicity, approximating cubicity is also as hard as boxicity (up to a factor of $\log n$). Also, they construct graph $G'_{\vec{P}}$ such that the threshold dimension of $G'_{\vec{P}}$ is the same as the dimension of poset $\vec{P}$, hence implying the hardness of approximating threshold dimension.

7 Conclusion and Open Problems

We have shown that simple techniques based on graph products are powerful tools in proving hardness of approximation. While some of these results are tight, some problems are still open.

In particular, it remains to close the gap between $d^{1/2-\epsilon}$ and $O(d)$ for the semi-induced matching problem on d-regular graphs. Also, there is a gap between $O(k)$-approximation (see [6]) and $k^{1/2-\epsilon}$-hardness for the k-hypergraph vertex pricing problem.

It is also interesting to further investigate the power of our techniques in proving hardness of approximation or even in other types of lower bounds. A potential starting point is to look at problems which share common structures to those studied in this paper. For example, UDP-MIN is a special case of the multi-user Stackelberg network pricing problem, so graph products can be used to prove the hardness of this problem. However, the approximability of the single-user version is still wide open as there is a large gap between $(2-\epsilon)$-hardness and $O(\log n)$-approximation algorithm. In fact, the proof of the $(2-\epsilon)$-hardness can be viewed as a reduction from the independent set problem, but we found no graph product techniques that can be used to boost the hardness further. (See, e.g., [8, 9] for detail.)

References

[1] Abhijin Adiga, Diptendu Bhowmick, and L. Sunil Chandran. The hardness of approximating the boxicity, cubicity and threshold dimension of a graph. *Discrete Appl. Math.*, 158(16):1719–1726, 2010.

[2] Abhijin Adiga, Diptendu Bhowmick, and L. Sunil Chandran. Boxicity and poset dimension. *SIAM J. Discrete Math.*, 25(4):1687–1698, 2011.

[3] Abhijin Adiga, L. Sunil Chandran, and Rogers Mathew. Cubicity, degeneracy, and crossing number. In *FSTTCS*, pages 176–190, 2011.

[4] Christoph Ambühl, Monaldo Mastrolilli, Nikolaus Mutsanas, and Ola Svensson. Precedence constraint scheduling and connections to dimension theory of partial orders. *Bulletin of the EATCS*, 95:37–58, 2008.

[5] Matthew Andrews, Julia Chuzhoy, Venkatesan Guruswami, Sanjeev Khanna, Kunal Talwar, and Lisa Zhang. Inapproximability of edge-disjoint paths and low congestion routing on undirected graphs. *Combinatorica*, 30(5):485–520, 2010.

[6] Maria-Florina Balcan and Avrim Blum. Approximation algorithms and online mechanisms for item pricing. *Theor. Comput.*, 3(1):179–195, 2007.

[7] Vincenzo Bonifaci, Peter Korteweg, Alberto Marchetti-Spaccamela, and Leen Stougie. Minimizing flow time in the wireless gathering problem. *ACM Transactions on Algorithms*, 7(3):33, 2011.

[8] Patrick Briest, Parinya Chalermsook, Sanjeev Khanna, Bundit Laekhanukit, and Danupon Nanongkai. Improved hardness of approximation for stackelberg shortest-path pricing. In *WINE*, pages 444–454, 2010.

[9] Patrick Briest, Martin Hoefer, and Piotr Krysta. Stackelberg network pricing games. *Algorithmica*, 62(3-4):733–753, 2012.

[10] Patrick Briest and Piotr Krysta. Buying cheap is expensive: Approximability of combinatorial pricing problems. *SIAM J. Comput.*, 40(6):1554–1586, 2011.

[11] Kathie Cameron. Induced matchings. *Discrete Appl. Math.*, 24(1-3):97–102, 1989.

[12] Kathie Cameron and Pavol Hell. Independent packings in structured graphs. *Math. Program.*, 105(2-3):201–213, 2006.

[13] Parinya Chalermsook, Julia Chuzhoy, Sampath Kannan, and Sanjeev Khanna. Improved hardness results for profit maximization pricing problems with unlimited supply. In *APPROX-RANDOM*, pages 73–84, 2012.

[14] L. Sunil Chandran, Mathew C. Francis, and Naveen Sivadasan. Boxicity and maximum degree. *J. Comb. Theory, Ser. B*, 98(2):443–445, 2008.

[15] L. Sunil Chandran and Naveen Sivadasan. Boxicity and treewidth. *J. Comb. Theory, Ser. B*, 97(5):733–744, 2007.

[16] William Duckworth, David Manlove, and Michele Zito. On the approximability of the maximum induced matching problem. *J. Discrete Algorithms*, 3(1):79–91, 2005.

[17] Khaled M. Elbassioni, Rajiv Raman, Saurabh Ray, and René Sitters. On the approximability of the maximum feasible subsystem problem with 0/1-coefficients. In *SODA*, pages 1210–1219, 2009.

[18] S. Even, O. Goldreich, S. Moran, and P. Tong. On the np-completeness of certain network testing problems. *Networks*, 14(1):1–24, 1984.

[19] Uriel Feige and Joe Kilian. Zero knowledge and the chromatic number. *J. Comput. Syst. Sci.*, 57(2):187–199, 1998.

[20] Stefan Felsner, Ching Man Li, and William T. Trotter. Adjacency posets of planar graphs. *Discrete Math.*, 310(5):1097–1104, 2010.

[21] F.S.Roberts. On the boxicity and cubicity of a graph. *Recent Progresses in Combinatorics*, pages 301–310, 1969.

[22] Guogang Gao and Xuding Zhu. Star-extremal graphs and the lexicographic product. *Discrete Math.*, 152(1-3):147–156, 1996.

[23] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[24] Zvi Gotthilf and Moshe Lewenstein. Tighter approximations for maximum induced matchings in regular graphs. In *WAOA*, pages 270–281, 2005.

[25] Venkatesan Guruswami, Jason D. Hartline, Anna R. Karlin, David Kempe, Claire Kenyon, and Frank McSherry. On profit-maximizing envy-free pricing. In *SODA*, pages 1164–1173. SIAM, 2005.

[26] Venkatesan Guruswami and Prasad Raghavendra. Hardness of solving sparse overdetermined linear systems: A 3-query pcp over integers. *TOCT*, 1(2), 2009.

[27] Richard Hammack, Wilfried Imrich, and Sandi Klavžar. *Handbook of product graphs*. Discrete Math. Appl. (Boca Raton). CRC Press, Boca Raton, FL, second edition, 2011.

[28] Johan Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *FOCS*, pages 627–636, 1996.

[29] Rajneesh Hegde and Kamal Jain. The hardness of approximating poset dimension. *Electron. Notes Discrete Math.*, 29:435–443, 2007.

[30] Chien-Chung Huang and Zoya Svitkina. Donation center location problem. In *FSTTCS*, pages 227–238, 2009.

[31] David S. Johnson. The NP-completeness column: An ongoing guide. *J. Algorithms*, 2(4):393–405, 1981.

[32] Changhee Joo, Gaurav Sharma, Ness B. Shroff, and Ravi R. Mazumdar. On the complexity of scheduling in wireless networks. *EURASIP J. Wireless Comm. and Networking*, 2010, 2010.

[33] Subhash Khot and Ashok Kumar Ponnuswami. Better inapproximability results for maxclique, chromatic number and min-3lin-deletion. In *ICALP (1)*, pages 226–237, 2006.

[34] Sandi Klavzar and Hong-Gwa Yeh. On the fractional chromatic number, the chromatic number, and graph products. *Discrete Math.*, 247(1-3):235–242, March 2002.

[35] Ravi Kumar, Uma Mahadevan, and D. Sivakumar. A graph-theoretic approach to extract storylines from search results. In *KDD*, pages 216–225, 2004.

[36] Eugene L. Lawler and Oliver Vornberger. The partial order dimension problem is NP-complete. Manuscript, 1981.

[37] Nathan Linial and Umesh V. Vazirani. Graph products and chromatic numbers. In *FOCS*, pages 124–128, 1989.

[38] Nikola Milosavljevic. On complexity of wireless gathering problems on unit-disk graphs. In *ADHOC-NOW*, pages 308–321, 2011.

[39] Sofya Raskhodnikova. Transitive-closure spanners: A survey. In *Property Testing*, pages 167–196, 2010.

[40] Paat Rusmevichientong. A non-parametric approach to multi-product pricing: Theory and application. *Ph. D. thesis, Stanford University*, 2003.

[41] Paat Rusmevichientong, Benjamin Van Roy, and Peter W. Glynn. A nonparametric approach to multiproduct pricing. *Oper. Res.*, 54:82–98, January 2006.

[42] E.R. Scheinerman and D.H. Ullman. *Fractional graph theory: a rational approach to the theory of graphs*. Wiley-Intersci. Ser. Discrete Math. Optim. Wiley, 1997.

[43] Walter Schnyder. Planar graphs and poset dimension. *Order*, 5:323–343, 1989.

[44] Walter Schnyder. Embedding planar graphs on the grid. In *SODA*, pages 138–148, 1990.

[45] Larry J. Stockmeyer and Vijay V. Vazirani. Np-completeness of some generalizations of the maximum matching problem. *Inf. Process. Lett.*, 15(1):14–19, 1982.

[46] Luca Trevisan. Non-approximability results for optimization problems on bounded degree instances. In *STOC*, pages 453–461, 2001.

[47] Luca Trevisan. CS294: PCP and Hardness of Approximation, Lecture 5, 2006.

[48] W.T. Trotter. *Combinatorics and Partially Ordered Sets: Dimension Theory*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2001.

[49] Mihalis Yannakakis. The Complexity of the Partial Order Dimension Problem. *SIAM J. Algebra Discr.*, 3(3):351–358, 1982.

[50] Michele Zito. Induced matchings in regular graphs and trees. In *WG*, pages 89–100, 1999.

Appendix

A Omitted Proofs from Section 6

A.1 Proof of Theorem 6.1

Theorem 6.1.(restated) *Consider an instance $G = (V_1 \cup V_2, E)$ of the bipartite semi-induced matching problem. Let $m = |V_1|$ and $n = |V_2|$. There is a polynomial time reduction that, for any $\beta > 0$, outputs an instance $\mathcal{A} = (A, \ell, \mu)$ of* MRFS *satisfying the following properties:*

- *(*SIZE*) Matrix A is an m-by-n matrix and $L = \max_{i \in [m]} \{\ell_i\} = (\beta n)^{O(m)}$.*
- *(*YES-INSTANCE*) There is a solution $\mathbf{x} \in \mathbb{R}^n_+$ that satisfies at least $\mathsf{im}(G)$ constraints in $\mathcal{A}$.*
- *(*NO-INSTANCE*) There is no solution $\mathbf{x} \in \mathbb{R}^n_+$ that "β-satisfies" more than $\mathsf{sim}(G)$ constraints in $\mathcal{A}$; i.e., $|\left\{i : \ell_i \le a_i^T \mathbf{x} \le \beta\mu_i\right\}| \le \mathsf{sim}(G)$ for all $\mathbf{x}$.*

Proof. For the sake of presentation, we represent the set of vertices of G as $V_1 = \{(u,1) : u \in [m]\}$ and $V_2 = \{(u,2) : u \in [n]\}$.

We define a linear system consisting of n variables $\{x_{(w,2)}\}_{w \in [n]}$ and the following m constraints:

$$\forall u \in [m], \quad (\beta n)^{3u-1} \le \sum_{w:(u,1)(w,2) \in E(G)} x_w \le (\beta n)^{3u} \tag{20}$$

Formally, for each $(u,w) \in [m] \times [n]$, define $a_{u,w} = 1$ if $(u,1)(w,2) \in E(G)$ and $a_{u,w} = 0$ otherwise. Then we create a constraint u for each vertex $(u,1) \in V_1$ as $\ell_u \le \sum_{(w,2) \in V_2} a_{u,w} x_w \le \mu_u$ where $\ell_u = (\beta n)^{3u-1}$ and $\mu_u = (\beta n)^{3u}$.

By the construction, the number of constraints is $m = |V_1|$, and the number of variables is $n = |V_2|$. Also, notice that $L = \max_{(u,1) \in V_1} \ell_u = (\beta n)^{O(m)}$. This proves the first property.

Let $\mathcal{M} = \{(u_i,1)(w_i,2) : i = 1, \ldots, r\}$ be an induced matching of size r in G. We can define the following solution for linear system $\mathcal{A}$: For each $i = 1, \ldots, r$, we have $x_{w_i} = \ell_{u_i}$, and $x_{w'} = 0$ for all other $w' \in [n]$. It suffices to show that a constraint u_i is satisfied for all $i \in [r]$. To see this, consider any constraint u_i, where $i \in [r]$. Only variables x_{w_j} with $(u_i,1)(w_j,2) \in E(G)$ participate in this constraint, and the only variable with positive value is $x_{w_i} = \ell_{u_i}$; otherwise, it would contradict the fact that $\mathcal{M}$ is an induced matching. This proves the second property.

Now, to prove the third property, assume that we have a solution $\mathbf{x}$ that β-satisfies r constraints, for some $0 < \beta \le |V(G)|$; i.e, there exists a subset $V_1^* \subseteq V_1$, denoted by $V_1^* = \{(u_1,1), \ldots, (u_r,1)\}$, such that

$$\forall (u_i,1) \in V_1^* \quad (\beta n)^{3u_i-1} \le \sum_{w:(u_i,1)(w,2) \in E(G)} x_w \le \beta(\beta n)^{3u_i} .$$

We note the following claim.

Claim A.1. *For any $(u_i,1) \in V_1^*$, there exists $w_i \in [n]$ such that $x_{w_i} \ge \ell_{u_i}/n$ and $(u_i,1)(w_i,2) \in E(G)$.*

Proof. Consider the constraint u_i, which involves variables $x_{w'}$ for all $w' \in [n]$ such that $(u_i,1)(w',2) \in E(G)$. Since there are at most n such variables and $\sum_{w':(u_i,1)(w',2) \in E(G)} x_{w'} \ge \ell_{u_i}$, one of the variables $x_{w'}$ must have a value of at least ℓ_{u_i}/n. Thus, $w_i = w'$ is the desired index, proving the claim. □

Next, we define a set of matching $\mathcal{M}$ by $\{(u_i,1)(w_i,2)\}_{i=1}^r$ where w_i is as in the above claim. It is not difficult to check that $\mathcal{M}$ is a matching: For any $u_i > u_j$, note that $x_{w_i} \geq \ell_{u_i}/n = (\beta n)^{3u_i-1}/n > \beta(\beta n)^{3u_j} = \beta\mu_{u_j} \geq x_{w_j}$; thus, $w_i \neq w_j$.

We define a total order σ on $V(G)$ as $\sigma(v) = v$ for all $(v,1) \in V_1$ and $\sigma(w) = m + w$ for all $(w,2) \in V_2$. We claim that $\mathcal{M}$ is a σ-semi-induced matching. To see this, assume that it is not. Then, by the definition of σ-semi-induced matching and the fact that G is bipartite, there must be some edge $(u_i,1)(w_j,2) \in E(G)$ for some i and j such that $u_i < u_j$. Observe that

$$\sum_{w':(u_i,1)(w',2)\in E(G)} x_{w'} \geq x_{w_j} \geq \ell_{u_j}/n = (\beta n)^{3u_j-1}/n > \beta(\beta n)^{3u_i} = \beta\mu_{u_i}.$$

This means that constraint u_i is violated by more than a factor of β, contradicting the assumption that $\mathbf{x}$ β-satisfies constraints corresponding to vertices in V_1^*. This proves the third claim and completes the proof of Theorem 6.1. □

A.2 Equivalence between semi-induced matching and maximum expanding sequence

In this section, we show that the maximum expanding sequence problem is in fact equivalent to the semi-induced matching problem.

Maximum Expanding Sequence. We are given an ordered collection of sets $\mathcal{S} = \{S_1,\ldots,S_m\}$ over the ground elements $\mathcal{U}$. An *expanding sequence* $\varphi = (\varphi(1),\ldots,\varphi(\ell))$ of length ℓ is a selection of sets $S_{\varphi(1)},\ldots,S_{\varphi(\ell)}$ such that, for all $j : 2 \leq j \leq \ell$, we have $\varphi(j-1) < \varphi(j)$ and $S_{\varphi(j)} \not\subseteq \bigcup_{j'<j} S_{\varphi(j')}$. Our objective is to compute an expanding sequence of maximum length.

Given an instance $(\mathcal{S},\mathcal{U})$ of the maximum expanding sequence problem, we denote by $\mathsf{OPT}(\mathcal{S},\mathcal{U})$ the optimal value of the instance. The following theorem shows that this problem is equivalent to the maximum semi-induced matching problem.

Theorem A.2. *Let $(\mathcal{S},\mathcal{U})$ be an instance of the maximum expanding sequence problem. Then there is a polynomial-time reduction that constructs an instance (G,σ) of the σ-semi-induced matching problem such that $\mathsf{sim}_\sigma(G) = \mathsf{OPT}(\mathcal{S},\mathcal{U})$. Conversely, given an instance (G,σ) of the semi-induced matching problem, we can construct $(\mathcal{S},\mathcal{U})$ such that $\mathsf{OPT}(\mathcal{S},\mathcal{U}) = \mathsf{sim}_\sigma(G)$.*

Proof. We only prove one direction of the reduction. It will be clear from the description that the converse also holds. Given an instance $(\mathcal{S},\mathcal{U})$ of the expanding sequence problem, we construct the bipartite graph $G = (V_1 \cup V_2, E)$ where $V_1 = \{(i,1) : i \in [|\mathcal{S}|]\}$ and $V_2 = \{(i,2) : i \in [|\mathcal{U}|]\}$. Each $S_i \in \mathcal{S}$ corresponds to $(i,1) \in V_1$ and each element $j \in \mathcal{U}$ corresponds to vertex $(j,2)$ in V_2. The set S_i contains j if and only if $(i,1)(j,2) \in E$, and finally the total order σ is defined as follows:

- $\sigma((i,1)) = i$ for all $0 \leq i \leq |\mathcal{S}| - 1$, and
- $\sigma((j,2)) = |\mathcal{S}| + j$ for all $0 \leq j \leq |\mathcal{U}| - 1$.

Notice that the total order σ put the order of vertices in V_1 before those in V_2, and the ordering of vertices in V_1 are ordered according to their corresponding sets.

We now claim that expanding sequences in $(\mathcal{S},\mathcal{U})$ are equivalent to σ-semi-induced matchings in G. For any expanding sequence, $S_{\varphi(1)},\ldots,S_{\varphi(\ell)}$, we define the σ-semi-induced matching as follows: For each $j = 1,\ldots,\ell$, we have an edge $(\varphi(j),1)(\psi(j),2)$ where $\psi(j)$ is defined as an arbitrary element in $S_{\varphi(j)} \setminus \left(\bigcup_{j'<j} S_{\varphi(j')}\right)$ (we know such element exists due to the property of expanding sequences). It is now easy to check that the set $\mathcal{M} = \{(\varphi(j),1)(\psi(j),2)\}$ is indeed a σ-semi-induced matching. □

A.3 Proof of Theorem 6.3

First, we describe our construction. We view the graph $G' = \mathrm{B}_e[G \vee H]$ as a bipartite graph $(V_1 \cup V_2, E)$ where $V_1 = \{(u, a, 1) : u \in V(G), a \in V(H)\}$ and $V_2 = \{(u, a, 2) : u \in V(G), a \in V(H)\}$. For convenience, we may think of vertices in $V(G)$ and $V(H)$ as integers (so that their ordering and arithmetic can be naturally done). For each vertex $(u, a, 2) \in V_2$, we have an item $I(u, a)$. So, $\mathcal{I} = \{I(u, a) : u \in V(G), a \in V(H)\}$, and hence $|\mathcal{I}| = |V(G)||V(H)|$. For each vertex $(u, a, 1) \in V_1$, we have n^{3a} consumers $\mathcal{C}(u, a) = \{c(u, a, r)\}_{r=1}^{n^{3a}}$, and each such consumer in this set has budget $B_{c(u,a,r)} = 1/n^{3a}$ and an associated set $S_{c(u,a,r)} = \{I(v, b) : (u, a, 1)(v, b, 2) \in E\}$. The final set of consumers is $\mathcal{C} = \bigcup_{u,a} \mathcal{C}(u, a)$. This completes the construction. The instance $(\mathcal{C}, \mathcal{I})$ here will be used as both Udp-Min and Smp instances.

We first show that the optimal revenue we receive from the above instance is at least the size of the maximum induced matching, for both Udp-Min and Smp. Let $\mathsf{OPT}_{\mathsf{UDP}}(\mathcal{C}, \mathcal{I})$ and $\mathsf{OPT}_{\mathsf{SMP}}(\mathcal{C}, \mathcal{I})$ denote the optimal value on instance $(\mathcal{C}, \mathcal{I})$ of Udp-Min and Smp, respectively.

Lemma A.3. *The followings hold for* Udp-Min *and* Smp*:*

- $\mathsf{OPT}_{\mathsf{UDP}}(\mathcal{C}, \mathcal{I}) \geq \mathsf{im}(G')$
- $\mathsf{OPT}_{\mathsf{SMP}}(\mathcal{C}, \mathcal{I}) \geq \mathsf{im}(G')$

Proof. Let $\mathcal{M}$ be an induced matching of cardinality K. First, for each $(u, a, 1)(v, b, 2) \in \mathcal{M}$, we set the price of $I(v, b)$ to $p(I(v, b)) = 1/n^{3a}$. Next, we set prices of the other items. For Udp-Min, we set their prices to ∞. For Smp, we set their prices to 0. It is clear that this price function is well-defined because $\mathcal{M}$ is a matching.

Now, we argue that the revenue that can be made from the price function p is K for both Udp-Min and Smp. It suffices to show that, for each pair (u, a) such that $(u, a, 1)$ is matched in $\mathcal{M}$, each consumer in the set $\mathcal{C}(u, a)$ pays the price of $1/n^{3a}$. Consider any edge $(u, a, 1)(v, b, 2) \in \mathcal{M}$. For Udp-Min, any consumer $c \in \mathcal{C}(u, a)$ has exactly one item $I(v, b) \in S_c$ with finite price of $1/n^{3a}$; otherwise, $\mathcal{M}$ would not be an induced matching. Similarly, for Smp, any consumer $c \in \mathcal{C}(u, a)$ has exactly one item in $I(v, b) \in S_c$ with non-zero price of $1/n^{3a}$. Therefore, the total profit made from consumers in $\mathcal{C}(u, a)$ is exactly one for both Udp-Min and Smp, and the lemma follows. □

The next lemma proves the upper bound of the revenue.

Lemma A.4. *The followings hold for* Udp-Min *and* Smp*:*

- $\mathsf{OPT}_{\mathsf{UDP}}(\mathcal{C}, \mathcal{I}) \leq 2\mathsf{sim}(G') + |V(G)|(|E(H)| + 1)$
- $\mathsf{OPT}_{\mathsf{SMP}}(\mathcal{C}, \mathcal{I}) \leq 2\mathsf{sim}(G') + O(|V(G)||E(H)|)$

Proof. Let p^* be any *optimal* price function for either Udp-Min or Smp, and let K be the revenue that we obtain from p^*. Our goal is to show that we can find a semi-induced matching of cardinality $K/2 - |V(G)|(|E(H)| + 1)$ in G'.

First, observe that, for any $u \in V(G), a \in V(H)$, all consumers in $\mathcal{C}(u, a)$ pay exactly the same price since they desire the same set of items and have the same budget. So, we will treat these consumers as a bundle and refer to index (u, a) as a representative of all consumers in $\mathcal{C}(u, a)$.

We need the following notion of tight index, which intuitively captures the consumers who spend a sufficiently large fraction of their budgets in the solution p^*.

Definition A.5 (Tight Index)**.** *We say that an index* (u, a) *is* tight *if consumers in* $\mathcal{C}(u, a)$ *pay between* $1/n^{3a+1}$ *and* $1/n^{3a}$ *(i.e., between* $1/n$ *and* 1 *proportion of their budgets).*

Claim A.6. *The number of tight indices is at least $K/2$.*

Proof. Assume for a contradiction that the number of tight indices is less than $K/2$. Consider a non-tight index (u,a) such that consumers in $\mathcal{C}(u,a)$ *can afford* to buy items:

- For Udp-Min, $\min_{(v,b):(u,a,1)(v,b,2)\in E(G')} p^*(I(v,b)) \in [0, 1/n^{3a+1})$.
- For Smp, $\sum_{(v,b):(u,a,1)(v,b,2)\in E(G')} p^*(I(v,b)) \in [0, 1/n^{3a+1})$.

We call these indices *feasible non-tight* indices.

Observe that we earn a profit of strictly larger than $K/2$ from consumers corresponding to these feasible non-tight indices in both Udp-Min and Smp, and this is because the price function p^* yields a revenue of K which only comes from either consumers with tight or feasible non-tight indices, and we get strictly less than $K/2$ for tight indices.

Now, define a new price function $p' = 2p^*$. By using the price function p', the revenue that we earn from the feasible non-tight indices will be twice. So, we would have revenue strictly more than K for both Udp-Min and Smp. This contradicts the optimality of p^*. □

Our goal is to "recover" a large semi-induced matching of G' from the tight indices. We will show that the set of "recoverable indices", which is large, is exactly the following set of *canonical tight indices*. Note that, in both Udp-Min and Smp, for any tight index (u,a), there must be an item $I(v,b)$ such that $(u,a,1)(v,b,2) \in E(G')$ (i.e., the item that consumers in $\mathcal{C}(u,a)$ want to buy) and $1/n^{3a+2} \le p^*(I(v,b)) \le 1/n^{3a}$ (i.e., $I(v,b)$ is expensive but affordable by consumers in $\mathcal{C}(u,a)$). We say that (u,a) is *canonical* if and only if $I(u,a)$ is the only such item. To be precise, the canonical tight index is defined as below.

Definition A.7 (Canonical Tight Index)**.** *We say that a tight index (u,a) is* canonical *if for any (v,b) such that $(u,a,1)(v,b,2) \in E(G')$, we have that $1/n^{3a+2} \le p^*(I(v,b)) \le 1/n^{3a}$ if and only if $u = v$ and $a = b$.*

First, we show that the number of canonical tight indices is large.

Claim A.8. *There are at least $K/2 - |V(G)|(|E(H)|+1)$ canonical tight indices.*

Proof. Let (u,a) be any non-canonical tight index. Recall that since (u,a) is tight, there must be an item $I(v,b)$ such that $(u,a,1)(v,b,2) \in E(G')$ and $1/n^{3a+2} \le p^*(I(v,b)) \le 1/n^{3a}$. Since (u,a) is non-canonical, it is *not* the case that both $u = v$ and $a = b$. Consequently, since $(u,a,1)(v,b,2) \in E(G')$, either $uv \in E(G)$ or $ab \in E(H)$. We say that (u,a) is *G-non-canonical* if $uv \in E(G)$ and *H-non-canonical* if $ab \in E(H)$. Note that every non-canonical tight index must be either G-non-canonical or H-non-canonical (or both).

First, we claim that the number of G-non-canonical tight indices is at most $|V(G)|$. In particular, we claim that for any $u \in V(G)$, there is *at most* one G-non-canonical tight index of the form $(u,a,1)$. To see this, let us assume for a contradiction that there are two G-non-canonical tight indices (u,a) and (u,a') for some $u \in V(G)$ and $a, a' \in V(H)$ such that $a < a'$.

For the case of Udp-Min, observe that since (u,a') is G-non-canonical and tight, there exists an index (v,b') such that (1) $uv \in E(G)$ and (2) $1/n^{3a'+2} \le p^*(I(v,b')) \le 1/n^{3a'}$. The first property implies that $(u,a,1)(v,b',2) \in E(G')$. Consequently, by the second property, consumers in $\mathcal{C}(u,a)$ pays *at most* $1/n^{3a'} < 1/n^{3a+1}$, contradicting the assumption that index (u,a) is tight.

For the case of Smp, observe that since (u,a) is G-non-canonical and tight, there exists an index (v,b) such that (1) $uv \in E(G)$ and (2) $1/n^{3a+2} \le p^*(I(v,b)) \le 1/n^{3a}$. The first property implies that $(u,a',1)(v,b,2) \in E(G')$. Consequently, by the second property, consumers in $\mathcal{C}(u,a)$ pays *at least* $1/n^{3a+2} > 1/n^{3a'}$, contradicting the assumption that index (u,a') is tight.

Secondly, we claim that the number of H-non-canonical tight indices is at most $|V(G)||E(H)|$. To see this, recall that for any (u,a) that is H-non-canonical and tight, there exists an index (v,b) such that (1) $ab \in E(H)$ and (2) $1/n^{3a+2} \le p^*(I(v,b)) \le 1/n^{3a}$. Observe that, by the first condition, any H-non-canonical tight index (u,a) must be in the following set $\Phi(u) = \bigcup_{a,b:\ ab \in E(H)} \{(u,a)\}$. Obviously, $|\Phi(u)| \le |E(H)|$. It follows that the number of H-non-canonical tight indices is at most $\sum_{u \in V(G)} |\Phi(u)| \le |V(G)||E(H)|$ as claimed.

Now, we have already shown that there are at most $|V(G)|(|E(H)|+1)$ non-canonical tight indices; since the number of tight indices is at least $K/2$ by the previous claim, we have that the number of canonical tight indices is at least $K/2 - |V(G)|(|E(H)|+1)$ as desired. $\square$

We finish the proof by showing that we can recover a large semi-induced matching from canonical tight indices.

Claim A.9. $\mathsf{sim}(G')$ *is at least the number of canonical tight indices. In other words,* $\mathsf{sim}(G') \ge K/2 - |V(G)|(|E(H)|+1)$.

Proof. Let $(u_1,a_1),(u_2,a_2)\ldots(u_t,a_t)$, for some t, be the canonical tight indices. Order them in such a way that $a_1 \le a_2 \le \ldots \le a_t$. Let $\mathcal{M} = \{(u_i,a_i,1)(u_i,a_i,2)\}_{i=1\ldots t}$.

For the case of Udp-Min, let σ be any total ordering of nodes in G' such that $\sigma(u_1,a_1,1) < \sigma(u_2,a_2,1) < \ldots < \sigma(u_t,a_t,1)$ and $\sigma(u_i,a_i,2) > \sigma(u_t,a_t,1)$ for all i. We claim that $\mathcal{M}$ is a σ-semi-induced matching in G'. In particular, we show that for any $i<j$, $(u_i,a_i,1)(u_j,a_j,2) \notin E(G')$. To see this, note that since (u_j,a_j) is canonical and tight, $1/n^{3a_j+2} \le p^*(I(u_j,a_j)) \le 1/n^{3a_j}$.

Consider two possible cases: either (1) $a_i < a_j$ or (2) $a_i = a_j$. In the first case, we have that $p^*(I(u_j,a_j)) \le 1/n^{3a_j} < 1/n^{3a_i+1}$. Thus, if $(u_i,a_i,1)(u_j,a_j,2) \in E(G')$, then consumers in $\mathcal{C}(u_i,a_i)$ will pay strictly less than $1/n^{3a_i+1}$, contradicting the fact that (u_i,a_i) is tight. In the second case, we have that $1/n^{3a_i+2} \le p^*(I(u_j,a_j)) \le 1/n^{3a_i}$. Thus, if $(u_i,a_i,1)(u_j,a_j,2) \in E(G')$, then (u_i,a_i) is not a canonical index, a contradiction. Since both cases lead to a contradiction, we have that $(u_i,a_i,1)(u_j,a_j,2) \notin E(G')$, and thus $\mathcal{M}$ is a σ-semi-induced matching as claimed.

For the case of Smp, let σ' be any total ordering such that $\sigma'(u_1,a_1,1) > \sigma'(u_2,a_2,1) > \ldots > \sigma'(u_t,a_t,1)$ and $\sigma'(u_i,a_i,2) > \sigma'(u_1,a_1,1)$ for all i. We claim that $\mathcal{M}$ is a σ'-semi-induced matching in G'. In particular, we show that for any $i>j$, $(u_i,a_i,1)(u_j,a_j,2) \notin E(G')$. To see this, note again that since (u_j,a_j) is canonical and tight, $1/n^{3a_j+2} \le p^*(I(u_j,a_j)) \le 1/n^{3a_j}$.

Consider two possible cases: either (1) $a_i > a_j$ or (2) $a_i = a_j$. In the first case, we have that $p^*(I(u_j,a_j)) \ge 1/n^{3a_j+2} > 1/n^{3a_i}$. Thus, if $(u_i,a_i,1)(u_j,a_j,2) \in E(G')$, then consumers in $\mathcal{C}(u_i,a_i)$ pay strictly more than $1/n^{3a_i}$, contradicting the fact that (u_i,a_i) is tight. In the second case, we have that $1/n^{3a_i+2} \le p^*(I(u_j,a_j)) \le 1/n^{3a_i}$. Thus, if $(u_i,a_i,1)(u_j,a_j,2) \in E(G')$, then (u_i,a_i) is not a canonical index, a contradiction. Since both cases lead us to a contradiction, we have that $(u_i,a_i,1)(u_j,a_j,2) \notin E(G')$, and thus $\mathcal{M}$ is a σ'-semi-induced matching as claimed. $\square$

The above claim completes the proof of the lemma. $\square$