Chalermsook, Parinya; Jiamjitrak, Wanchote Po; Orgo, Ly

On Finding Balanced Bicliques via Matchings

Published in:
Graph-Theoretic Concepts in Computer Science - 46th International Workshop, WG 2020, Revised Selected Papers

DOI:
10.1007/978-3-030-60440-0_19

Published: 01/01/2020

Document Version
Peer reviewed version

Please cite the original version:
Chalermsook, P., Jiamjitrak, W. P., & Orgo, L. (2020). On Finding Balanced Bicliques via Matchings. In I. Adler, & H. Müller (Eds.), *Graph-Theoretic Concepts in Computer Science - 46th International Workshop, WG 2020, Revised Selected Papers* (pp. 238-247). (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 12301 LNCS). SPRINGER. https://doi.org/10.1007/978-3-030-60440-0_19

On Finding Balanced Bicliques via Matchings*

Parinya Chalermsook[1,2], Wanchote Po Jiamjitrak[1,3], and Ly Orgo[1,4]

[1] Aalto University, Finland
[2] parinya.chalermsook@aalto.fi
[3] wanchote.jiamjitrak@aalto.fi
[4] ly.orgo@aalto.fi

Abstract. In the Maximum Balanced Biclique Problem (MBB), we are given an n-vertex graph $G = (V, E)$, and the goal is to find a balanced complete bipartite subgraph with q vertices on each side while maximizing q. The MBB problem is among the first known NP-hard problems, and has recently been shown to be NP-hard to approximate within a factor $n^{1-o(1)}$, assuming the *Small Set Expansion hypothesis* [Manurangsi, ICALP 2017]. An $O(n/\log n)$ approximation follows from a simple brute-force enumeration argument. In this paper, we provide the first approximation guarantees beyond brute-force: (1) an $O(n/\log^2 n)$ efficient approximation algorithm, and (2) a parameterized approximation that returns, for any $r \in \mathbb{N}$, an r-approximation algorithm in time $\mathsf{exp}\left(O(\frac{n}{r \log r})\right)$. To obtain these results, we translate the subgraph removal arguments of [Feige, SIDMA 2004] from the context of finding a clique into one of finding a balanced biclique. The key to our proof is the use of matching edges to guide the search for a balanced biclique.

1 Introduction

The Maximum Balanced Biclique (MBB) problem is among the oldest and most fundamental NP-hard graph problems. It was stated to be NP-hard (without proof) in Garey and Johnson's book [12]; a proof is provided, for instance, in [14]. In this problem, we are given an n-vertex graph G, and we are interested in finding a balanced complete bipartite subgraph with q vertices on each side while maximizing the value of q. Since the problem is NP-hard, the main theoretical interest so far has been on approximation algorithms [18, 11], parameterized algorithms [16], and parameterized approximation [4]. All results so far have been on the negative side, suggesting that MBB is very highly intractable. First, in terms of approximation algorithms, Manurangsi [18] showed that the problem is NP-hard to approximate within a factor of $n^{1-o(1)}$ assuming the *Small Set Expansion (SSE)* hypothesis and that $\mathbf{NP} \not\subseteq \mathbf{BPP}$. The other hardness result, somewhat incomparable to Manurangsi's is shown by Khot [15], that MEB does not admit n^{ϵ} approximation, for some $\epsilon > 0$, unless $\mathbf{NP} \subseteq \mathbf{BPTIME}(2^{n^{\Omega(1)}})$. On the parameterized algorithm side, a recent remarkable result of Lin [16]

* This manuscript is part of Ly Orgo's master thesis at Aalto University.

has shown that MBB is $\mathbf{W}[1]$-hard, therefore not admiting an FPT algorithm unless $\mathbf{FPT} = \mathbf{W}[1]$. Finally, MBB does not even admit any $o(\mathcal{OPT})$ FPT approximation algorithm [4] assuming the *Gap Exponential-Time Hypothesis (Gap-ETH)*, where $\mathcal{OPT}$ is the number of vertices in the optimal biclique.

The focus of this paper is on approximation algorithms. The aforementioned negative results suggest that MBB is among the "highly intractable problems" (those that do not admit $n^{1-o(1)}$ approximation), likely to be in the same ballpark as the clique, independent set, induced matching, and graph coloring problem. All these problems admit an $O(n/\log n)$ approximation algorithm via brute-force enumeration techniques [9, 6]. For maximum clique or graph coloring problems, a lot of attention in the approximation algorithms community has been on obtaining any algorithm that beats these brute-force algorithms. In the context of maximum clique and independent set problems, many LP/SDP (as well as combinatorial) approaches have been devised, that achieve guarantees beyond trivial algorithms (see for instance [10, 2, 13] and references therein). However, such results do not exist at all in the context of MBB.

1.1 Our results and techniques

In this paper, we provide the first set of approximation algorithms on MBB whose approximation guarantees are asymptotically better than brute-force. Our first result is an efficient algorithm that runs in polynomial time. Throughout the paper, we use *size* of a balanced biclique to denote the number of vertices on one side. In particular, the size of the complete bipartite graph with q vertices on each side ($K_{q,q}$) is q.

Theorem 1. *There is an $O(n/\log^2 n)$ polynomial time approximation algorithm for MBB.*

Our second result is a parameterized approximation that gives a tradeoff between approximation ratio and running time.

Theorem 2. *For any $r \in \mathbb{N}$, there exists an r-approximation algorithm running in time $\exp\left(O(\frac{n}{r \log r})\right)$.*

Now we give a high-level discussion that highlights our main technical ideas. Let $\mathcal{OPT}$ denote the maximum value of q such that $K_{q,q}$ exists in G. Notice that, when $\mathcal{OPT} < n/\log^2 n$, we are immediately done since we can return a single edge and it would be $n/\log^2 n$ approximation. Therefore, we may assume that $\mathcal{OPT} \geq n/\log^2 n$. Our main result shows that we can efficiently find a biclique containing roughly $\tilde{\Omega}(\log^2 n)$ vertices on each side[5], and this would also be an $O(n/\log^2 n)$ approximation.

[5] Here, we use the convention that $\tilde{\Omega}$ hides asymptotically smaller terms.

Enumeration by Vertices: By a standard brute-force enumeration argument, one can easily find a biclique of size $\Omega(\log n/\log\log n)$. We explain this algorithm as an intuitive starting point. Assume that the graph $G = (A \cup B, E)$ is bipartite [6] with $|A| = |B| = n$. Fix an optimal biclique Q in G. We partition A into $A_1 \cup A_2 \cup \ldots \cup A_\ell$ arbitrarily where each set A_j contains $|A_j| = \lceil \log^3 n \rceil$ vertices each (that is, we have $\ell = n/\lceil \log^3 n \rceil$ sets.) Recall that $\mathcal{O}PT \geq n/\log^2 n$. By averaging argument, there must be a "good" set A_j that contains $\frac{\log^3 n}{\log^2 n} = \log n$ vertices from the optimal biclique Q, that is, $|A_j \cap V(Q)| \geq \log n$. We then enumerate all subsets $X \subseteq A_j : |X| = (a \log n/\log\log n)$; for each X, let $C_X = \{v \in B : N_G(v) \supseteq X\}$ be the neighbors that can be included to form a biclique with X (that is, $G[C \cup C_X]$ is a biclique). We return any pair of sets (X, C_X) such that $|C_X| \geq |X|$. The running time of this procedure is $\frac{n}{\lceil \log^3 n \rceil}\binom{\log^3 n}{(\log n/\log\log n)} \leq (\log^3 n)^{\log n/\log\log n} = \mathsf{poly}(n)$.

Feige's Subgraph Removal. The above enumeration trick has been used several times for many problems (including clique and biclique) in the literature. It provides immediately a search procedure for a clique/biclique of size $\Theta(\frac{\log n}{\log\log n})$ whenever $\mathcal{O}PT \geq n/(\log^{O(1)} n)$. For clique (as well as independent set), Feige [10] "augmented" a subgraph removal procedure on top of this vertex enumeration procedure, so that his algorithm returns a clique of size $\Omega((\log n/\log\log n)^2)$ instead. Unfortunately, the above-mentioned natural idea of vertex enumeration is not quite compatible with Feige's subgraph removal arguments.

New Idea: Enumeration by Matching Edges. This is where our new observation comes in handy. We perform a "matching-edge" enumeration instead of a vertex enumeration. We explain the intuition of our proof by describing another procedure that finds a biclique containing $\Omega(\log n/\log\log n)$ vertices on each side. The main benefit of matching-edge enumeration over vertex enumeration is its versatility, which allows us to use Feige's subgraph removal trick [10] to derive our desired result.

Again we fix an n-vertex m-edge bipartite graph $G = (A \cup B, E)$ and an optimal biclique Q in G. Since we focus on the case $\mathcal{O}PT \geq n/(\log^2 n)$, we have that $|E(Q)| = \mathcal{O}PT^2 \geq n^2/(\log^4 n)$. We partition the edges in E into at most n matchings, that is, $E = E_1 \cup E_2 \cup \ldots \cup E_n$, where each set E_j is a matching. By dismissing all small sets, it is easy to see that there exists some larger set E_j such that $|E_j| \geq 16(\log^5 n)$ and $|E_j \cap E(Q)| \geq |E_j|/2(\log^4 n)$. We again divide such set E_j into $E_{j,1} \cup E_{j,2} \cup \ldots \cup E_{j,s}$ such that $\lceil 2\log^5 n \rceil \leq |E_{j,\alpha}| \leq \lceil 4 \log^5 n \rceil$ for all $\alpha = 1, 2, \ldots, s$. By averaging (using the fact that $|E_j \cap E(Q)| \geq |E_j|/(2\log^4 n)$), there exists $E_{j,\alpha}$ such that $|E_{j,\alpha} \cap E(Q)| \geq \frac{2\log^5 n}{2\log^4 n} \geq \log n$. We can enumerate all size-$(\log n/\log\log n)$ subsets $M \subseteq E_{j,\alpha}$ in time $(\log^5 n)^{O(\log n/\log\log n)} = \mathsf{poly}(n)$. Each such subset M is a matching, and we can check whether it induces a biclique

[6] See Lemma 1 for a simple formal proof that it suffices to focus on the case of bipartite graphs.

in G. This concludes an algorithm that finds a biclique with $\Omega(\log n/\log\log n)$ vertices on each side, based on matching-edge enumeration.

In Section 3, we show how to implement Feige's subgraph removal procedure on top of the matching-edge enumeration procedure. The presentation there is self-contained and does not rely on the discussion in this section.

1.2 Further Related Results

Prior to Khot [15], Feige and Kogan showed that unless $\mathbf{NP} \subseteq \mathbf{DTIME}(2^{n^{\frac{3}{4}+o(1)}})$, the problem does not admit $2^{\Omega((\log n)^{\Omega(1)})}$ approximation. Due to interest on application sides, there have been several heuristics [21, 20] proposed for MBB, but none of them give any theoretical guarantee on the approximation factor.

The Maximum Edge Biclique (MEB) problem is very similar to the MBB problem, except for the fact that MEB aims to maximize the number of edges in a (possibly not balanced) biclique. Similarly to MBB, MEB is known to be $n^{1-o(1)}$ hard to approximate under the same complexity-theoretic assumptions [18]. Assuming more standard complexity assumptions, the problem is known to be n^{ϵ} hard to approximate [1].

The covering variants of biclique problems (called *minimum biclique cover*) are better understood from approximation perspectives: There is a $n^{1-o(1)}$ hardness result assuming $\mathbf{P} \neq \mathbf{NP}$ [5] and some non-trivial algorithms exist in the contexts of both approximation and parameterized algorithms [7].

The trade-off between an approximation factor and the running time has recently received attention. See e.g. [3, 8, 19, 2] and the references therein.

2 Preliminaries

This paper follows standard notation in graph theory. Given a graph $G = (V, E)$, denote by $N_G(v)$ the neighboring vertices of v in G (excluding v). For $q \in \mathbb{N}$, denote by $K_{q,q}$ the complete bipartite graph with q vertices on each side. For any subset $S \subseteq V$, denote by $G[S]$, the induced subgraph on S. We sometimes abuse notation and use, for each edge set $F \subseteq E$, $G[F]$ to represent $G[V(F)]$.

An s-edge coloring of graph $G = (V, E)$ is a partition of E into $E_1 \cup E_2 \cup \ldots \cup E_s$ where each E_i is a matching in G. We will use the following edge coloring theorem of König (see, for instance, [17]).

Theorem 3. *Given a bipartite graph $G = (A \cup B, E)$ where each node has a degree of at most Δ, there exists a Δ-edge coloring of G that can be computed efficiently.*

We show that we can focus only on designing approximation algorithms for bipartite graphs.

Lemma 1. *If there is an α-approximation algorithm for MBB in bipartite graphs, then there is an $O(\alpha)$ approximation for MBB in general graphs.*

Proof. We turn a general graph $G = (V_G, E_G)$ into a bipartite graph $H = (L \cup R, E_H)$ as follows: For each vertex $v \in V_G$, add the vertex into either L or R independently with probability $1/2$. Next, we keep only the edges between L and R, that is, $E_H = \{(u, v) : u \in L, v \in R\}$.

Let $A \cup B$ be an optimal complete bipartite subgraph in G where $|A| = |B| = \mathcal{O}PT$. Let M be a perfect matching in $G[A \cup B]$, so $|M| = \mathcal{O}PT$. We say that an edge $e = (u, v) \in M$ is good if $u \in L$ and $v \in R$. Let $M' \subseteq M$ be the set of good edges. Notice that, $\mathbb{E}[|M'|] = |M|/4$ and that the vertices of M' induce a biclique in H. Therefore, in expectation, the biclique $A \cup B$ appears in H as a biclique of size at least $\mathcal{O}PT/4$ on each side, so the presumed α-approximation would be able to return a biclique of size $\mathcal{O}PT/4\alpha$ in expectation.

To obtain a deterministic algorithm, notice that the above proof only relies on pairwise independence of the choice of random bits. □

3 Our Algorithms

3.1 Subgraph Removal implies Approximation Algorithms

We prove an analogue of Feige's subgraph removal procedure in the context of MBB. By simple calculation, it implies both of our algorithmic results.

Theorem 4. *Given a graph $G = (V, E)$ with a maximum balanced biclique of size n/z, for each $t = O(\frac{n}{z^5})$, there exists an algorithm that runs in time $z^{O(t)} poly(n)$ and finds a balanced biclique of size $q = \Theta(t \log_z \frac{n}{t})$.*

Now we show that this theorem implies both Theorem 1 and Theorem 2. These proofs are standard (see [10, 2]) and are only presented here for completeness of exposition.

Corollary 1. *For any $r \in \mathbb{N}$, there exists an r-approximation algorithm running in time $\exp\Big(O(\frac{n}{r \log r})\Big)$.*

Proof. We are given a bipartite graph $G = (A \cup B, E)$. If $\mathcal{O}PT \le n/(\log^2 r)$, we are done since we can enumerate all subsets $S \subseteq A$ of size $n/(r \log^2 r)$ (and this would be an r-approximation) in time

$$\binom{n}{n/(r \log^2 r)} \le (er \log^2 r)^{n/(r \log^2 r)} \le 2^{O(n/(r \log r))}$$

Otherwise, we have that $\mathcal{O}PT = n/z$ for $z \le \log^2 r$. Choose $t = \frac{n}{r \log r \log z}$ so that $z^{O(t)} = 2^{O(\frac{n}{r \log r})}$. It is easy to check that $t \le O(n/z^5)$ for sufficiently large n. Theorem 4 gives us a biclique with at least $\Omega(t \cdot \log_z(n/t)) = \Omega(\frac{n \log r}{r \log r \log^2 z}) = \Omega(\frac{n}{r \log^2 z})$ nodes on each side. The approximation factor obtained is:

$$\frac{n/z}{\Omega(\frac{n}{r \log^2 z})} = O(r)$$

□

Corollary 2. *There is a polynomial time $O(n/\log^2 n)$ approximation algorithm.*

Proof. If $\mathcal{O}PT \leq n/(\log^2 n)$, we are immediately done. Otherwise, assume that $z = O(\log^2 n)$. Let $t = \log_z n$, we use Theorem 4 to find a balanced biclique $K_{q,q}$ where $q = \Theta(\log_z^2 n)$ (so this algorithm runs in polynomial time). Therefore, the approximation factor is $\mathcal{O}PT/q \leq O(\frac{n \log^2 z}{z \log^2 n}) \leq O(n/\log^2 n)$. □

3.2 Proof of Theorem 4

We now describe our algorithm. It has two steps. In the first step, we perform a pre-processing by removing vertices whose degrees are too large compared to the size of the optimal biclique. This step will allow us to apply König's edge coloring theorem, decomposing the graph into a union of disjoint matchings.

Step 1: Degree Reduction We are given an n-vertex bipartite graph $G' = (V', E')$ that contains an optimal biclique Q of size $p = n/z$; define a parameter k, such that $p^2 = m/k$ is the number of edges in the optimal biclique.

This step is summarized in the following lemma.

Lemma 2. *There is an efficient algorithm that produces a graph $G = (V, E)$ such that:*

- $|V(G)| \geq n/2$
- *Each vertex in G has degree at most $2pk$.*
- *There exists a biclique containing $p/2$ vertices on each side of G.*

Proof. Whenever there is a vertex v whose degree is more than $2pk = 2\sqrt{mk}$, we remove v and all edges incident to v (but we keep vertices in $N_G(v)$.) Notice that we would remove at most $\frac{1}{2}\sqrt{m/k} = \frac{p}{2}$ vertices from the graph (since we have at most m edges), and therefore at least $p/2$ vertices remain on each side of Q after such removals. This completes the proof. □

Step 2: An Analogue of Feige's argument Assume we are given a graph $G = (A \cup B, E)$ that satisfies the conditions in Lemma 2. Let Q be the optimal biclique in G containing at least $p/2$ vertices on each side (therefore containing at least $m/4k$ edges). This choice of optimal clique Q is fixed throughout the execution of the algorithm.

Definition 1. *We say that a subset of edges E'' is poor if $|E'' \cap E(Q)| < |E''|/8k$.*

Definition 2. *Let Q' be a balanced biclique. An edge e is said to be* ***consistent*** *with Q' if $Q' \cup \{e\}$ induces a biclique and the endpoints of e is are disjoint from $V(Q)$, that is, if Q' is $K_{r,r}$, we must have that $Q' \cup \{e\}$ induces a $K_{r+1,r+1}$. Denote by $\mathcal{C}E(Q')$ the set of edges that are consistent with biclique Q'.*

Now we present our algorithm. The algorithm has multiple phases, and each phase consists of many iterations that keep growing the set of matching edges F whose vertices induce a biclique.

In the beginning, $E' \leftarrow E$. In each phase, we start from edge set $E'' \leftarrow E'$ and $F \leftarrow \emptyset$. In each iteration, if we have that $|E''| \leq 10^6 pk^3 t$, then we say that the **phase terminates successfully and the algorithm terminates**. Otherwise, we partition E'' into $2kp$ subsets $E''_1 \cup E''_2 \cup \ldots \cup E''_{2kp}$ where each E''_j is a matching. We say that E''_j is large if $|E''_j| \geq 16kt$; otherwise, we say that E''_j is small. Consider each large set E''_j and partition it further (arbitrarily) into $E''_{j,1} \cup E''_{j,2} \cup \ldots \cup E''_{j,\ell(j)}$ where each set has size at least $16kt$ and at most $32kt$. A *good matching* is a subset $M \subseteq E''_{j,\alpha} : |M| = t$ that satisfies two conditions: (i) $G[M]$ is a biclique and (ii) the set of consistent edges $E_M = \mathcal{C}E(G[M]) \cap E''$ is sufficiently large, that is, $|E_M| \geq |E''|/8k - (5pkt)$. If a good matching exists in some subset $E''_{j,\alpha}$, we update $F \leftarrow F \cup M$, $E'' \leftarrow E_M$, and then start the new iteration. Otherwise, if no good matching is found, we claim that E'' is a poor subset (proof provided below), the **phase terminates unsuccessfully**, and we start a new phase with $E' \leftarrow E' \setminus E''$.

Analysis of Running Time

A phase either ends with a poor subset of edges removed from the graph (when it is unsuccessful) or it ends with a collection of matching edges F (when it is successful). Each poor subset is a subset of size at least one, so there can be at most m unsuccessful phases.

Lemma 3. *The running time of each phase is at most $z^{O(t)} n^{O(1)}$.*

Proof. In each iteration, each set $E''_{j,\alpha}$ has size at most $32kt$ and we enumerate all subsets of size t inside it. There are at most n iterations in each phase. So, the total running time would be at most:

$$\binom{32kt}{t} n^{O(1)} \leq \left(\frac{32ekt}{t}\right)^t n^{O(1)} \leq k^{O(t)} n^{O(1)}.$$

Since $k = \frac{z^2 m}{n^2} < z^2$, then $k^{O(t)} = z^{O(t)}$.

The Size of Bicliques

Now we proceed to show that at some point, the algorithm would terminate successfully and return a relatively large biclique.

Lemma 4. *If no good matching is found in a phase, then E'' is a poor subset.*

Proof. Assume, by contrapositive, that E'' is not poor. Then $|E'' \cap E(Q)| \geq |E''|/8k$. Notice that the number of edges in the small sets E''_j is at most $16kt \cdot (2kp) \leq |E''|/16k$. Therefore, at least $|E''|/16k$ edges in Q appear in one of the large sets. Since $|E'' \cap E(Q)|/|E''| \geq 1/16k$, by averaging argument, we have that there exists a large set $E''_{j,\alpha}$ for which $|E''_{j,\alpha} \cap E(Q)|/|E''_{j,\alpha}| \geq 1/16k$ which implies that $|E''_{j,\alpha} \cap E(Q)| \geq |E''_{j,\alpha}|/16k \geq t$. Let M be an arbitrary size-t subset of $E''_{j,\alpha} \cap E(Q)$. Notice that M is good: Firstly, $G[M]$ is a biclique. Moreover, any edge e in $E'' \cap E(Q)$ that is not sharing a vertex with any edge in M, is consistent with $G[M]$. There are at least $|E''|/8k - 5pkt$ such edges, because the number of edges that have two vertices in M is less than t^2 and the number of edges that have one vertex in M is at most $\Delta(G) \cdot 2t \leq 4kpt$. Since $t \leq p$, then $t^2 + 4kpt < 5kpt$. □

So, the above lemma implies that whenever a subset of edges is removed from the graph, that subset must be poor. The following lemma says that, at some point, a poor subset would not exist anymore.

Lemma 5. *Let $\hat{E}_1, \hat{E}_2, \ldots, \hat{E}_\ell$ be a collection of poor subsets removed from the phases. Then, $|\bigcup_i \hat{E}_i \cap E(Q)| < m/8k$*

Proof. This follows from the fact that $|\hat{E}_i \cap E(Q)| < |\hat{E}_i|/8k$. Summing over all i gives us the desired bound. □

Corollary 3. *At the beginning of each phase, we have that $|E' \cap E(Q)| \geq m/8k$.*

Proof. Since after the degree reduction we have at least $m/4k$ edges in $E' \cap E(Q)$, then by Lemma 5, we have at least $m/4k - m/8k = m/8k$ edges in $E(Q)$ left in each phase. □

With this, we know that the unsuccessful phase cannot remove too many edges from the optimal solution. Now, we argue that the result returned by a successful phase is a matching F that induces a biclique and it has the desired size.

Observation 5 *Let F_1 be a matching such that $G[F_1]$ is a biclique. Let $F_2 \subseteq \mathcal{CE}(G[F_1])$ be another matching such that $G[F_2]$ is also a biclique. Then $G[F_1 \cup F_2]$ is a biclique containing $|F_1| + |F_2|$ vertices on each side.*

This observation implies that the result returned by the algorithm induces a biclique. Its size is equal to the product of t with the number of iterations in that phase. The following lemma will finish the proof.

Lemma 6. *A successful phase runs for at least $\Omega(\log_z\left(\frac{m}{t}\right))$ iterations.*

Proof. Notice that in the same phase, each iteration, that starts with E'', proceeds to the next iteration (starting with E_M) on the condition that the number of remaining edges is at least $|E_M| \geq |E''|/8k - 5pkt \geq |E''|/16k$. At the beginning of the phase, there are at least $m/8k$ edges in E', and the stopping condition is when $|E''| \leq 10^6 pk^3 t$. Therefore, we can proceed for at least $\log_{16k}\left(\frac{m}{10^7 k^4 pt}\right)$ iterations. Since $m = \Theta(p^2 k)$ and $k = \frac{z^2 m}{n^2} \leq z^2$, then the number of iterations is $\Omega(\log_k\left(\frac{m}{t}\right)) \geq \Omega(\log_z\left(\frac{m}{t}\right))$. □

4 Discussions and Open Problems

In this paper, we present approximation algorithms for MBB whose guarantee is better than that of brute-force. One obvious open question is to match the $O(n/\log^3 n)$ approximation of clique, which would put biclique and clique in the same ballpark. A truly interesting direction is to study the power of semi-definite programs for bicliques. While there are many such algorithms for cliques and coloring, we do not have them for any biclique problem (including maximum edge biclique or biclique covering problems).

Finally, as discussed in [10], an interesting aspect of the subgraph removal algorithm is its connection to algorithmic Ramsey theory. In particular, the poor subgraph detection algorithm can be seen as a constructive Ramsey-type argument (please refer to discussions in Feige's paper for more detail.) While standard Ramsey arguments (i.e. clique v.s. independent set) have found their applications in theoretical computer science, ours is perhaps the first algorithmic result of Ramsey-type theorem for balanced bicliques.

Acknowledgement: This project is supported by European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 759557) and by Academy of Finland Research Fellowship, under grant number 310415. Ly Orgo has been supported by European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 759557).

References

1. Christoph Ambühl, Monaldo Mastrolilli, and Ola Svensson. Inapproximability results for maximum edge biclique, minimum linear arrangement, and sparsest cut. *SIAM Journal on Computing*, 40(2):567–596, 2011.
2. Nikhil Bansal, Parinya Chalermsook, Bundit Laekhanukit, Danupon Nanongkai, and Jesper Nederlof. New tools and connections for exponential-time approximation. *Algorithmica*, 81(10):3993–4009, 2019.
3. Édouard Bonnet, Michael Lampis, and Vangelis Th Paschos. Time-approximation trade-offs for inapproximable problems. *Journal of Computer and System Sciences*, 92:171–180, 2018.
4. Parinya Chalermsook, Marek Cygan, Guy Kortsarz, Bundit Laekhanukit, Pasin Manurangsi, Danupon Nanongkai, and Luca Trevisan. From Gap-ETH to FPT-inapproximability: Clique, dominating set, and more. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 743–754. IEEE, 2017.
5. Parinya Chalermsook, Sandy Heydrich, Eugenia Holm, and Andreas Karrenbauer. Nearly tight approximability results for minimum biclique cover and partition. In *European Symposium on Algorithms*, pages 235–246. Springer, 2014.
6. Parinya Chalermsook, Bundit Laekhanukit, and Danupon Nanongkai. Independent set, induced matching, and pricing: Connections and tight (subexponential time) approximation hardnesses. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 370–379. IEEE, 2013.

7. Sunil Chandran, Davis Issac, and Andreas Karrenbauer. On the parameterized complexity of biclique cover and partition. In *11th International Symposium on Parameterized and Exact Computation (IPEC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.
8. Marek Cygan, Lukasz Kowalik, Marcin Pilipczuk, and Mateusz Wykurz. Exponential-time approximation of hard problems. *arXiv preprint arXiv:0810.4934*, 2008.
9. Marek Cygan, Łukasz Kowalik, and Mateusz Wykurz. Exponential-time approximation of weighted set cover. *Information Processing Letters*, 109(16):957–961, 2009.
10. Uriel Feige. Approximating maximum clique by removing subgraphs. *SIAM Journal on Discrete Mathematics*, 18(2):219–225, 2004.
11. Uriel Feige and Shimon Kogan. Hardness of approximation of the balanced complete bipartite subgraph problem. *Dept. Comput. Sci. Appl. Math., Weizmann Inst. Sci., Rehovot, Israel, Tech. Rep. MCS04-04*, 2004.
12. Michael R Garey and David S Johnson. *Computers and intractability*, volume 29. WH freeman New York, 2002.
13. Eran Halperin. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. *SIAM Journal on Computing*, 31(5):1608–1623, 2002.
14. David S Johnson. The NP-completeness column: an ongoing guide. *Journal of Algorithms*, 6(3):434–451, 1985.
15. Subhash Khot. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM Journal on Computing*, 36(4):1025–1071, 2006.
16. Bingkai Lin. The parameterized complexity of k-biclique. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 605–615. SIAM, 2014.
17. László Lovász and Michael D Plummer. *Matching theory*, volume 367. American Mathematical Soc., 2009.
18. Pasin Manurangsi. Inapproximability of maximum edge biclique, maximum balanced biclique and minimum k-cut from the small set expansion hypothesis. In *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.
19. Pasin Manurangsi and Luca Trevisan. Mildly exponential time approximation algorithms for vertex cover, balanced separator and uniform sparsest cut. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
20. Yiyuan Wang, Shaowei Cai, and Minghao Yin. New heuristic approaches for maximum balanced biclique problem. *Information Sciences*, 432:362–375, 2018.
21. Bo Yuan, Bin Li, Huanhuan Chen, and Xin Yao. A new evolutionary algorithm with structure mutation for the maximum balanced biclique problem. *IEEE transactions on cybernetics*, 45(5):1054–1067, 2014.