Appendix

A Proof of the NP-hardness of the MBS problem

In this section, we prove the NP-hardness of the MBS problem by demonstrating a polynomial reduction from maximum clique search(MCP), which is a well-known NP-complete problem, to the maximum k-biplex search problem with k=1 [47].

It is worth noticing that [47] set the relation equation between α and α' to $\alpha' = \frac{1}{2}\alpha^3 + \frac{3}{2}\alpha^2 - \alpha$, which led to errors in the values of various expressions following cases: (1) In case 1, $|\mathbb{E}(\mathbb{G}(L \cup R)| \neq \frac{1}{2}\alpha^3 + \frac{3}{2}\alpha^2 - \alpha$(error α'), which affects the correctness of proof that if there exists a clique with α vertices in G then there exists a 1-biplex with α' edges in $\mathbb{G}$; (2) In case 2.1, it causes $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha'$ gets an error value. Similarly, in case 2.2, it also makes $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha'$ to be an error value. These make it impossible to guarantee that if G doesn't have a clique with α vertices, $\mathbb{G}$ also have no 1-biplex with α' edges. To avoid the above problem, we reset the correct relationship equation between α and α' in the following proof.

We first define two decision problems as follows.

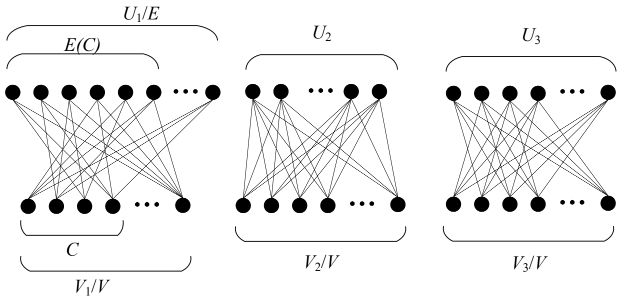
- CLIQUE: For a given general graph $G = (V, E)$ and a positive integer α, is there a clique consisting of at least α vertices?
- BIPLEX: For a given bipartite graph $\mathbb{G} = (\mathbb{U} \cup \mathbb{V}, \mathbb{E})$ and two positive integers k and α', is there a k-biplex containing at least α' edges?

Let $G = (V, E)$ be a graph and α be a given parameter, which together form the inputs for an instance of the CLIQUE problem. W.l.o.g., we assume that $\alpha = \frac{1}{2}|V|$ is a positive integer. We can achieve this in the following inflation strategies without affecting the result of the CLIQUE instance:

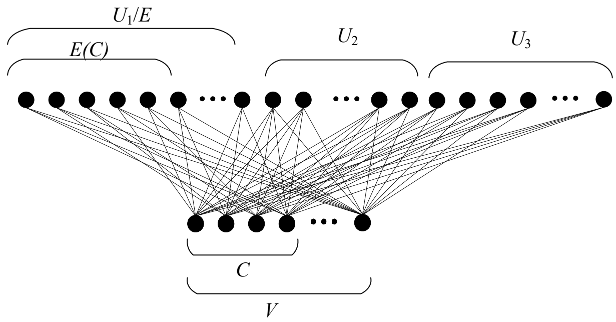
- In case $\alpha < \frac{1}{2}|V|$, we add a new vertex v into V that connect to each vertex in V. Clearly, every clique must include v. This operation can increase α by 1 but increase $\frac{1}{2}|V|$ by only 0.5.
- In case $\alpha > \frac{1}{2}|V|$, we add a new vertex v into V that does not connect to any vertex in V. Clearly, every clique must not include v. This operation can not increase α but increase $\frac{1}{2}|V|$ by 0.5.

Repeating the above two inflation strategies, we have $\alpha = \frac{1}{2}|V|$ finally.

Now, we construct the instance of BIPLEX (with $k = 1$). First, we construct a bipartite graph $G_1 = (U_1 \cup V_1, E_1)$ based on G. In G_1, V_1 consists of the nodes in V and U_1 consists of the edges in E. If a node v in V isn't the endpoint of edge e in E, we create a connection between v in V_1 and e in U_1. Therefore, we have



(a) Three graphs constructed from a MCP instance



(b) The merged graph

Fig. 22: Illustration of the NP-hardness proof

$U_1 = E$, $V_1 = V$, and $E_1 = \{(v, e)|v \in V, e \in E, v \notin e\}$. Second, we construct a biclique $G_2 = (U_2 \cup V_2, E_2)$ by G. To be specific, $V_2 = V$, $|U_2| = \frac{1}{2}\alpha(\alpha - 5)$. Third, we construct a 1-biplex $G_3 = (U_3 \cup V_3, E_3)$ by G. Specifically, we have $V_3 = V$, $|U_3| = |V| = 2\alpha$, and $E_3 = \{(v_i, u_j)|v_i \in V_3, u_i \in U_3, i \neq j\}$. These three graphs G_1, G_2 and G_3 are shown in Fig. 22(a).

By merging the three graphs, we obtain the input bipartite graph $\mathbb{G} = (\mathbb{U} \cup \mathbb{V}, \mathbb{E})$ of the instance of BIPLEX, which is shown in Fig. 22(b), and we have $\mathbb{U} = U_1 \cup U_2 \cup U_3, \mathbb{V} = V, \text{and} \mathbb{E} = E_1 \cup E_2 \cup E_3$. Here, we set another input α' of the instance of BIPLEX to be $\alpha^3 - \alpha^2 - \alpha$. we have $\alpha' = \alpha^3 - \alpha^2 - \alpha$. Because $|U_2| \geq 0$, we stipulate $\alpha \geq 5$, a condition readily met through the employment of inflationary strategies. As analyzed above, the construction of G_1, G_2, G_3, and $\mathbb{G}$ can be completed in polynomial time.

We then prove that it has a clique with at least α vertices from G iff there is a 1-biplex with at least α' edges from $\mathbb{G}$. Two cases need to be taken into account. **Case 1:** G has a clique $G(C)$ with α vertices, i.e., $C \subseteq V$ and $|C| = \alpha$. In this case, we can derive a 1-biplex $\mathbb{G}(L \cup R)$ where $L = V \backslash C$ and $R = E(C) \cup U_2 \cup U_3$. In G_1, for each $v \in L$ and e in $E(C)$, $v \notin e$, so we have $\mathbb{G}(L \cup E(C))$ is fully connected. In G_2, we have each vertex v in V connect all vertex in U_2 and $\mathbb{G}(L \cup U_2)$ is also fully connected. In G_3, for each vertex $v \in L$, v only loses 1 connection to vertices in U_3. Similarly, each vertex $u \in U_3$ also disconnects one vertex from L. Considering the above, $\mathbb{G}(L \cup R)$ satisfies the definition of 1-biplex. It holds that $|\mathbb{E}(\mathbb{G}(L \cup E(C))|=|L| \times |E(C)|=\alpha \times \frac{1}{2}\alpha(\alpha-1)$, $|\mathbb{E}(\mathbb{G}(L \cup U_2))|=|L| \times |U_2|=\alpha \times \frac{1}{2}\alpha(\alpha-5)$, $|\mathbb{E}(\mathbb{G}(L \cup U_3))|=|L| \times |U_3|=\alpha \times (2\alpha-1)$. In summary, $|\mathbb{E}(\mathbb{G}(L \cup R)| = \alpha^3 - \alpha^2 - \alpha$ which is exactly α'.

Case 2: G does not have a clique with at least α vertices. If we can demonstrate that there does not exist a 1-biplex with at least α' edges in $\mathbb{G}$, the proof is finished. Let $\mathbb{G}(L \cup R)$ be an arbitrary 1-biplex in $\mathbb{G}$ such that $L \subseteq \mathbb{U}$ and $R \subseteq \mathbb{V}$. We complete the proof by demonstrating that $|\mathbb{E}(\mathbb{G}(L \cup R))| < \alpha'$. First, considering the definition of 1-biplex, we divide R into two disjoint parts $R_0 \cup R_1$, i.e., $d(v) = |L|$ for $v \in R_0$ and $d(v) = |L| - 1$ for $v \in R_1$.

For R_0, we have: (1) In G_1, it comprises all vertices from $E(G(V \backslash L)) \subseteq E$ as every edge in $E(G(V \backslash L))$ has no endpoint in L, which is based on the construction of G_1; (2) In G_2, it encompasses all vertices from U_2 based on the construction of G_2; and (3) In G_3, it includes $|V \backslash L|$ vertices from U_3 ($L = \{v_1, ..., v_{|L|}\}$, vertices in $\{u_{|L|+1}, ..., u_{2\alpha}\} \subseteq U_3$ would connect all vertices from L based on the construction of G_3).

For R_1, each vertex $v \in L$ can only disconnect one vertex in R_1. $|R_1|$ achieve its upper bound $|L|$ if each vertex $v \in L$ disconnects a different vertex, so we have $|R_1| \leq |L|$.

In summary, we have $|\mathbb{E}(\mathbb{G}(L \cup R))| = |L| \times |R_0| + (|L|-1) \times |R_1| \leq |L| \times (|E(G(V \backslash L))| + |U_2| + |V \backslash L|) + (|L|-1) \times |L|$.

Let $x = |L|$, it is obviously that $0 \leq x \leq 2\alpha$. Then, we further consider the following two cases:

Case 2.1: $\alpha < x \leq 2\alpha$. Let $y = x - \alpha$ ($0 < y \leq \alpha$). We have $|L| = \alpha + y$ and $|V \backslash L| = \alpha - y$. If $G(V \backslash L)$ is a clique, $|E(G(V \backslash L))| = \frac{1}{2}|V \backslash L|(|V \backslash L| - 1)$, so we have $|E(G(V \backslash L))| \leq \frac{1}{2}|V \backslash L|(|V \backslash L| - 1)$. According to Equation (1), it has $\mathbb{E}(\mathbb{G}(L \cup R))| \leq (\alpha+R)[\frac{1}{2}(\alpha-R)(\alpha-R-1)+\frac{1}{2}\alpha^2-\frac{5}{2}\alpha+2\alpha-(\alpha+R)]+(\alpha+R-1)(\alpha+R)$ which reduces to $\mathbb{E}(\mathbb{G}(L \cup R))| \leq \alpha^3 - \alpha^2 - \alpha - \frac{1}{2}\alpha y^2 + \frac{1}{2}y^3 - \frac{1}{2}\alpha y + \frac{1}{2}y^2 - y$. Therefore, $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha' \leq \frac{1}{2}y[(y+1)(y-\alpha)-2]$. It is clearly that $\frac{1}{2}y[(y+1)(y-\alpha)-2] < 0$ since $0 < y \leq \alpha$. Hence, we can get $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha' < 0$, and $|\mathbb{E}(\mathbb{G}(L \cup R))| < \alpha'$.

Case 2.2: $0 \leq x \leq \alpha$. Let $y = \alpha - x$ ($0 \leq y \leq \alpha$). We have $|L| = \alpha - y$ and $|V \backslash L| = \alpha + y$. we know two following conditions: (1)$|V \backslash L| \geq \alpha$ and G does not have a clique with at least α vertices; (2) if we remove a vertex from a clique C in G to decrease its size by 1, we only need to remove at least one edge. It is easy to verify that $|E(G(V \backslash L))| < \frac{1}{2}|V \backslash L|(|V \backslash L| - 1) - y$ since we need remove at least y edges to guarantee that G does not have a clique with at least α vertices. According to Equation (1), we have $|\mathbb{E}(\mathbb{G}(L \cup R))| < (\alpha-y)[\frac{1}{2}(\alpha+y)(\alpha+y-1) - y + \frac{1}{2}\alpha^2 - \frac{5}{2}\alpha + 2\alpha - (\alpha - y)] + (\alpha-y-1)(\alpha-y)$ which reduces to $|\mathbb{E}(\mathbb{G}(L \cup R))| < \alpha^3 - \alpha^2 - \alpha - \frac{1}{2}\alpha y^2 - \frac{1}{2}\alpha y - \frac{1}{2}y^3 + \frac{3}{2}y^2 + y$. Then we have $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha' < \frac{1}{2}y[-y^2 + (3-\alpha)y + 2 - \alpha]$. It is clearly that $\frac{1}{2}y[-y^2 + (3-\alpha)y + 2 - \alpha] < 0$ since $0 < y \leq \alpha$ and $\alpha \geq 5$. So we can get $|\mathbb{E}(\mathbb{G}(L \cup R))| - \alpha' < 0$. Therefore, we have $|\mathbb{E}(\mathbb{G}(L \cup R))| < \alpha'$.

In summary, from Cases 1 and 2, it has a clique with at least α vertices from G iff there is a 1-biplex with at least α' edges from $\mathbb{G}$. Through the polynomial reduction from maximum clique search(MCP) to MBS, we prove the NP-hardness of MBS.

B Proof of the Inapproximability of the MBS problem

In this section, we prove the inapproximability of the maximum biplex problem by contradiction.

Lemma 1 *The maximum k-biplex search problem is inapproximable in polynomial time within a factor $n^{1-\epsilon}$, for all $\epsilon > 0$, unless P=NP.*

Proof Let *OPT(MCP)*$=\alpha$ be the optimal objective value for the MCP problem and *APX(MCP)*$=\beta$ be the approximate value. Since the MCP problem is an Optimization Problem, we have $\alpha \geq \beta$. According to Appendix A in this paper, if MBS has an optimal objective value *OPT(MBS)*$=\mathbb{G}(L \cup R)$ where $L = V \backslash C$ and $R = E(C) \cup U_2 \cup U_3$. From the reduction in Appendix A, we have *OPT(MBS)* $= \alpha^3 - \alpha^2 - \alpha$. Similarly, the approximate value of MBS is $\beta = \beta^3 - \beta^2 - \beta$. Suppose there exists a polynomial time algorithm $\mathcal{A}$ approximates the MBS problem with a factor of $\delta = n^{1-\epsilon} (\forall \epsilon > 0)$. Since MBS is also an Optimization Problem, we have $\frac{OPT(MBS)}{APX(MBS)} \geq \delta$ and

$$\delta APX(MBS) \leq OPT(MBS)$$

$$\Rightarrow \delta\beta^3 - \delta\beta^2 - \delta\beta \leq \alpha^3 - \alpha^2 - \alpha$$

$$\Rightarrow \delta \frac{\beta^2 - \beta - 1}{\alpha^2 - \alpha - \alpha} \leq \frac{\alpha}{\beta}.$$

It is clear that $\alpha, \beta \geq 3$ (the smallest clique is a triangle), we can get that $\frac{\beta^2-\beta-1}{\alpha^2-\alpha-\alpha} < 1$, and $\delta \times \frac{\beta^2-\beta-1}{\alpha^2-\alpha-\alpha}$ is smaller than δ. This implies the approximation ratio of MCP ($\frac{\alpha}{\beta}$) can achieve a threshold less than $n^{1-\epsilon}$, which contradicts the theorem that "For all $\epsilon > 0$, it is NP-hard to approximate MAX CLIQUE (MCP) within a factor $n^{1-\epsilon}$" from [21, 53].

As analyzed above, the maximum k-biplex search problem is inapproximable in polynomial time within a factor $n^{1-\epsilon}$ for all $\epsilon > 0$, unless P=NP. Therefore, this lemma holds.

C Additional Proofs

Claim 1 *Symmetric BK branching, like the BK branching, will not affect the accuracy of the results of MBS.*

Proof Similar to the BK branch, the symmetric BK branch also looks at all candidate subsets and can gain exact results. Suppose that $C = \{v_1, v_2, \cdots, v_{|C|}\}$. BK branching adds node v_i to Q, removes $\{v_1, \cdots, v_{i-1}\}$ from subspace $(Q \cup C)$, and creates $|C|$ branches. Symmetric BK branching removes each node v_i from subspace $(Q \cup C)$, adds $\{v_1, \cdots, v_{i-1}\}$ to Q, and creates $|C|+1$ branches.

The operation of the two branching methods to produce the 1^{st} through $|C|^{th}$ branches are symmetric, the only difference being that the $(|C|+1)^{th}$ branch. BK branching removes the entire set of C from the subspace to create $(|C|+1)^{th}$ branch that has no productive value and is therefore not created, whereas the symmetric BK branch adds all nodes of the set of C to Q to obtain $(|C|+1)^{th}$ branch that makes sense. For the same search space, any branch generated by BK branching has a corresponding symmetric branch in the branches generated by the symmetric BK branching.

As analyzed above, symmetric BK branching does not miss checking all possible cases of vertex sets, and it can also ensure the accuracy of the results of MBS.

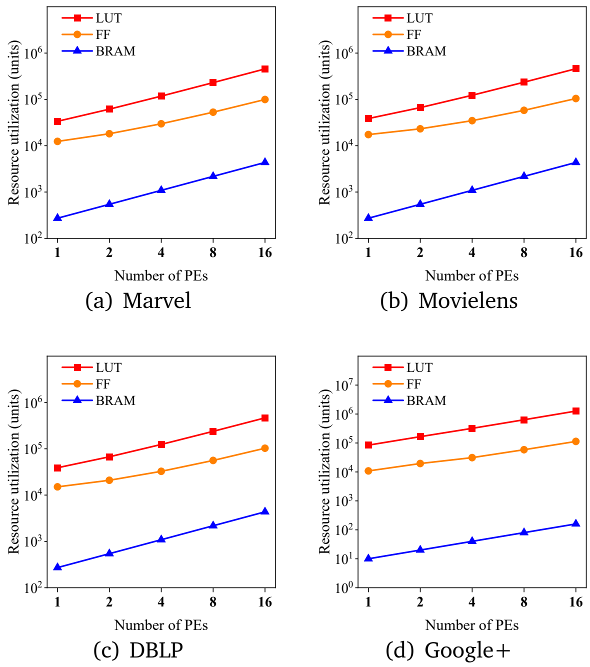
Claim 2 *The refinement of the graph into a (s, s)-core contributes to boosting the search performance, and will not affect the accuracy of the results of MBS.*

Proof In this paper, s represents the minimum number of nodes in U_P and V_P of all k-biplex P, i.e, $s \leq \min\{|U_P|, |V_P|\}$. For each node $v \in V_P$, we have $d(v) \geq |U_P| - k \geq s - k$. Similarly, for node $u \in U_P$, it holds $d(u) \geq |V_P| - k \geq s - k$. According to the definition of (x, y)-core (Definition 3), we have that the maximum k-biplex is always contained in a (s, s)-core. Therefore, instead of the given graph, searching the results of MBS from corresponding (s, s)-core can also gain accurate results of MBS. In this way, it can boost search performance by reducing the search space.

D Additional Experiments

The hardware resources utilization of FPS. In this experiment, we evaluate The consumption of hardware resources about BRAMs(Block RAM), LUTs(Look-Up Table), and FFs(Flip Flop). In this paper, DSPs (Digital Signal Processor) are not used and their consumption is not evaluated.

Fig. 23 shows the consumption of hardware resources of the FPS algorithm over the four real datasets, Marvel, Movielens, DBLP, and Google+, by varying the number of PEs. With the increase of PEs, LUT, FF, and BRAM all grow. This means that it requires much more hardware resources.



(a) Marvel (b) Movielens (c) DBLP (d) Google+

Fig. 23: The consumption of hardware resources.