Learning to Bound for Maximum Common Subgraph Algorithms

Buddhi W. Kothalawala
The Australian National University, Australia

Henning Koehler
Massey University, New Zealand

Qing Wang
The Australian National University, Australia

Abstract

Identifying the maximum common induced subgraph (MCIS) between two graphs is an NP-hard problem. The McSplit algorithm is a prominent method for solving the MCIS problem and uses a branch-and-bound (BnB) framework. Several extensions have been developed to improve McSplit's branching strategy using reinforcement learning (RL). However, the effectiveness of the bounding strategy is crucial for efficiently pruning unpromising branches and speeding up the search. This research presents a stricter upper bound based on an analysis of McSplit's partitions and introduces an RL approach that activate it in the most promising areas of the search.

2012 ACM Subject Classification Theory of computation → Branch-and-bound; Mathematics of computing → Combinatorial optimization

Keywords and phrases Combinatorial Search, Branch and Bound, Graph Theory

Digital Object Identifier 10.4230/LIPIcs.CP.2025.20

Supplementary Material *Software (Source Code)*: `https://github.com/BuddhiWathsala/mcsplit-dsb`

Funding This research was supported partially by the Australian Government through the Australian Research Council's Discovery Projects funding scheme (project DP210102273).

Acknowledgements We are thankful to Dr Muhammad Farhan for helpful discussions and feedback.

1 Introduction

Graphs are widely used in real-world applications due to their ability to model complex structures [2, 5, 4, 1]. Identifying common patterns between such graphs is a key challenge in graph theory. The MCIS problem involves finding two induced subgraphs from given graphs G and H that are isomorphic and contain the most vertices [3].

Various exact algorithms have been developed to solve the MCIS problem, particularly the McSplit algorithm [7], which uses heuristics for branching and bounding to enhance performance. Recent developments include RL enhancements like McSplit+LL [8] and McSplit+DAL [6], which aim to optimize vertex pair selection (i.e., branching) in McSplit. Unlike the existing RL extensions, we proposes a new approach to enhance upper bound estimation of the McSplit algorithm. The key contributions include: 1) A *tighter upper bound* using graph structural properties, which improves pruning efficiency while maintaining exactness and 2) A *dynamic RL framework* which activates the proposed bound computation in most potential areas in the search.

The 31st International Conference on Principles and Practice of Constraint Programming/28th International Conference on Theory and Applications of Satisfiability Testing, Doctoral Program 2025.
Editor: Maria Garcia de la Banda; Article No. 20; pp. 20:1–20:4
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 A New Upper Bound

We propose a tighter upper bound for the MCIS problem compared to McSplit's bound, which reduces the search space for BnB algorithms.

Let G be a simple, graph with vertices $V(G)$ and edges $E(G)$. For a vertex subset $V' \subseteq V(G)$, $G[V']$ denotes the *induced subgraph* of G using V'. Consider a mapping $M = \{(v_1, u_1), \ldots, (v_k, u_k)\}$, where $V_k = \{v_1, \ldots, v_k\} \subseteq V(G)$ and $U_k = \{u_1, \ldots, u_k\} \subseteq V(H)$. Here, $G[V_k]$ and $H[U_k]$ induced subgraphs are isomorphic. The McSplit algorithm partition the vertices of G and H (not in M) based on their connections to the mapped vertices. This partitioning ensures that after selecting a new pair (v_{k+1}, u_{k+1}), the subgraphs $G[V_{k+1}]$ and $H[U_{k+1}]$ remain isomorphic. Each partition is referred to as a *bidomain*, denoted as $\langle V_l, U_l \rangle$ and set of all bidomains denoted as P_{GH}.

In the McSplit algorithm, they use maximum size of $|V_l|$ and $|U_l|$ to find the maximum number of possible vertex pairs to be added to the current subgraph from a bidomain $\langle V_l, U_l \rangle$. The bound calculation of McSplit is too generous. For instance, if $G[V_l]$ is a clique of five vertices (K_5) and $H[U_l]$ is the complement graph ($\bar{K_5}$), the size of their maximum common subgraphs is one, which is significantly smaller than $\min(|V_l|, |U_l|)$ estimated in the bound. So, the proposed bound uses degree sequences of subgraphs $G[V_l]$ and $H[U_l]$ to estimate the maximum common subgraphs, rather than just the sizes $|V_l|$ and $|U_l|$ as in McSplit because degree sequences offer more structural insight than merely considering graph sizes.

Let $M^*(G, H)$ denote a maximum-cardinality mapping between two graphs G and H, and $Deg(G)$ the sequence of vertex degrees in G in ascending order and $d_G(v)$ denote the degree of a vertex v in G. The degree sum of G is defined as $d(G) = \sum_{i=1}^{n} d_G(v_i)$. We use $\bar{G}$ to denote the *complement graph* of a graph G. We derive the following.

▸ **Theorem 1.** Let $|V(G)| \leqslant |V(H)|$ and $Deg(G) = (d_G(v_1), \ldots, d_G(v_n))$. If

(1) $\sum_{i=1}^{k} d_G(v_i) - \sum_{j=k+1}^{n} d_G(v_j) > d(H)$, or

(2) $\sum_{j=n-k-1}^{n} d_{\bar{G}}(v_j) - \sum_{i=1}^{n-k} d_{\bar{G}}(v_i) > d(\bar{H})$

then $|M^*(G, H)| \leqslant k - 1$, where $d_{\bar{G}}(v_i) = |V(G)| - d_G(v_i) - 1$ for $i = 1, \ldots, n$.

Based on Theorem 1, we define the *dividing number* k of graphs G and H as the smallest number in $[1, n]$ satisfying Case (1) or Case (2). It can be shown that Cases (1) and (2) cannot both hold and only occur for $k > n/2$.

The bound in Theorem 1 is tighter than McSplit's. We define the *bound gap* δ_{GH} as the difference between McSplit's bound and that in Theorem 1. If a dividing number k exists then $\delta_{GH} = n - k + 1$, otherwise $\delta_{GH} = 0$.

For any bidomain $\langle V_l, U_l \rangle \in P_{GH}$, δ (the subscript is omitted) refers to the bound gap of $G[V_l]$ and $H[U_l]$. This gives us the following new bound where we call it as *degree sequence bound* (DSB): $UB_{dsb} \leftarrow |M| + \sum_{\langle V_l, U_l \rangle \in P_{GH}} min(|V_l|, |U_l|) - \delta$.

3 Learning-based Bounding Heuristic

Applying the proposed bound all the time is inefficient as the computation time of the proposed bound is higher than the McSplit bound. So, we suggest incorporating an RL agent that determine when to apply the tighter bound to maximize pruning efficiency by utilizing the new bound in search areas with high pruning potential, .

Let $\mathcal{P}(\cdot)$ represent powersets and ϕ a property of bidomains. A ϕ-*activation function* is defined as $\lambda : \mathcal{P}(V(G)) \times \mathcal{P}(V(H)) \rightarrow \{active, inactive\}$, where $\lambda(V', U') = active$ if $\langle V', U' \rangle$ satisfies property ϕ. At each branching step, P_{GH} includes a subset of active bidomains,

$P^\dagger_{GH} = \{\langle V_l, U_l\rangle \in P_{GH} \mid \lambda(V_l, U_l) = active\}$. The reward for actions on P_{GH} is based on the estimated bound gap between McSplit and the new bound for active bidomains: $R(V_l, U_l) = \delta_{G[V_l]H[U_l]}$.

Each bidomain has value function, $S(V_l, U_l)$ that is initialized to 1 and updated with a historical average weighted by $\alpha \in [0, 1]$: $S(V_l, U_l) \leftarrow (1 - \alpha)S(V_l, U_l) + \alpha R(V_l, U_l)$. For all active bidomains, the score for $S(P_{GH})$ is defined as: $S(P_{GH}) \leftarrow \sum_{\langle V_l, U_l\rangle \in P^\dagger_{GH}} S(V_l, U_l)$. At each branching step, the algorithm checks the condition, $UB - S(P_{GH}) \leqslant |incumbent|$ to decide on computing new bounds. If true, the new bounds are computed, and rewards and value scores are updated; otherwise, McSplit is used without changes.

4 Experiments

Experiments demonstrate that our approach outperforms McSplit and its variants in overall execution time, due to significant branch reduction. By applying our new bound, McSplit solves 3.41% more cases than the previous version, and identifies larger common subgraphs for unsolved instances. We also show that the bound closely approximates the maximum reduction achievable by any bound operating on bidomains alone.

References

1 Leman Akoglu, Hanghang Tong, and Danai Koutra. Graph-based anomaly detection and description: A survey. *Data Mining and Knowledge Discovery*, 29, 04 2014. `doi:10.1007/s10618-014-0365-y`.

2 Hans-Christian Ehrlich and Matthias Rarey. Maximum common subgraph isomorphism algorithms and their applications in molecular science: a review. *WIREs Computational Molecular Science*, 1(1):68–79, 2011.

3 Ruth Hoffmann, Ciaran McCreesh, and Craig Reilly. Between subgraph isomorphism and maximum common subgraph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

4 Xiuzhen Huang, Jing Lai, and Steven Jennings. Maximum common subgraph: Some upper bound and lower bound results. *BMC bioinformatics*, 7 Suppl 4:S6, 02 2006. `doi:10.1186/1471-2105-7-S4-S6`.

5 Takeshi Kawabata and Haruki Nakamura. 3d flexible alignment using 2d maximum common substructure: Dependence of prediction accuracy on target-reference chemical similarity. *Journal of Chemical Information and Modeling*, 54(7):1850–1863, 2014. PMID: 24895842. `arXiv:https://doi.org/10.1021/ci500006d`, `doi:10.1021/ci500006d`.

6 Yanli Liu, Jiming Zhao, Chu-Min Li, Hua Jiang, and Kun He. Hybrid learning with new value function for the maximum common induced subgraph problem. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4):4044–4051, Jun. 2023. URL: `https://ojs.aaai.org/index.php/AAAI/article/view/25519`, `doi:10.1609/aaai.v37i4.25519`.

7 Ciaran McCreesh, Patrick Prosser, and James Trimble. A partitioning algorithm for maximum common subgraph problems. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 712–719, 2017. `doi:10.24963/ijcai.2017/99`.

8 Jianrong Zhou, Kun He, Jiongzhi Zheng, Chu-Min Li, and Yanli Liu. A strengthened branch and bound algorithm for the maximum common (connected) subgraph problem.

In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2022.