

Enhancing Max- k -Cut Solvers through Bio-Inspired Approaches

Pietro De Angeli^{1*}[0009-0003-2168-1247], Giorgia Gabardi^{1*}[0009-0006-8985-3588], Alberto Vendramini^{1*}[0009-0006-9700-1289], Erik Nielsen¹[0009-0007-8176-5396], Stefano Genetti¹[0009-0004-2417-0319], and Giovanni Iacca¹[0000-0001-9723-1830]

Department of Information Engineering and Computer Science
University of Trento, Italy
{pietro.deangeli, giorgia.gabardi, alberto.vendramini}@studenti.unitn.it
{erik.nielsen, stefano.genetti, giovanni.iacca}@unitn.it

Abstract. This work investigates the integration of bio-inspired optimization techniques with modern learning-based approaches for solving the Max- k -Cut problem. Building upon high-quality solutions produced by the *Relax-Optimize-and-Sample* (ROS) framework, a recent state-of-the-art Graph Neural Network (GNN)-based method for Max- k -Cut, we explore the effectiveness of two bio-inspired paradigms: Genetic Algorithm, Ant Colony Optimization, along with some of their variants. The proposed methods exploit ROS outputs as strong initial solutions, effectively combining data-driven learning with evolutionary and swarm-based search strategies. Experimental evaluations are conducted on a representative subset of instances from the Gset benchmark suite for multiple values of k . Results show that all considered bio-inspired approaches consistently improve upon the baseline ROS solutions, with hybrid variants incorporating local search yielding the largest gains. In particular, the multispecies competitive coevolution framework demonstrates superior performance by maintaining population diversity and mitigating premature convergence. These findings highlight the benefits of combining GNN-based solvers with bio-inspired optimization to achieve robust and high-quality solutions for large-scale combinatorial optimization problems.

Keywords: Graph Pattern Recognition · Evolutionary Computation · Swarm Intelligence · Relax-Optimize-and-Sample · Max- k -Cut

1 Introduction

Pattern recognition on graph-structured data is increasingly important in domains such as social network analysis [2, 3, 14, 7, 6, 10], biological interaction

* These authors contributed equally to this research.

modelling [15], Very Large Scale Integration (VLSI) design [1], large-scale clustering [22], and neural network sparsification [5]. In many of these applications, the underlying task can be formulated as a graph partitioning problem, where the objective is to identify meaningful structural patterns and relationships between graph components.

One of the most important problems on graphs, from both a theoretical and application perspective, is the Max- k -Cut problem, where the goal is to partition the vertices of a graph into k disjoint subsets, maximizing the total weight of edges crossing between different subsets. Recent advances in Graph Neural Networks (GNNs) [4] have enabled learning-based approaches capable of producing high-quality solutions quickly, but the resulting assignments may still leave room for improvement, especially on large and difficult instances.

Taking inspiration from adaptive mechanisms observed in natural systems, in this work, we propose using bio-inspired optimization techniques to provide complementary capabilities, such as exploration and diversity preservation, to a GNN-based solver. Specifically, we propose a hybrid pipeline where a GNN-based solver, *Relax-Optimize-and-Sample* (ROS) [19], provides a strong starting point, which is then refined by bio-inspired search based on either using Genetic Algorithm (GA) or Ant Colony Optimization (ACO), both in their base form and in hybrid variants with local search.

From a pattern-recognition perspective, the proposed methodology can be interpreted as a hybrid adaptive recognition system operating on graph-structured patterns. The GNN-based ROS component performs representation learning and initial pattern assignment; the subsequent bio-inspired refinement stage acts as an adaptive mechanism that improves partition quality through distributed exploration and local adaptation. This hybridisation reflects recent trends in bio-inspired pattern recognition systems, where evolutionary and swarm-based processes complement gradient-based learning.

Our contribution is an empirical assessment of ROS-seeded bio-inspired refinement on a representative subset of the Gset benchmark [21] for $k \in \{2, 3, 10\}$, highlighting which combinations provide the most consistent improvements and how diversity-preserving mechanisms affect performance as k increases. Across all tested (instance, k) pairs, our bio-inspired refinements consistently improve upon ROS, effectively turning a state-of-the-art learning-based solver into an even stronger hybrid optimizer. This supports the broader view that bio-inspired search can be productively combined with learning-based methods to further refine high-quality solutions.

To guide our investigation, we formulate the following research questions:

- RQ1: Can bio-inspired optimization methods consistently improve the solution quality obtained by the ROS framework for the Max- k -Cut problem?
- RQ2: How do different bio-inspired optimization methods compare in their ability to refine ROS-generated solutions?

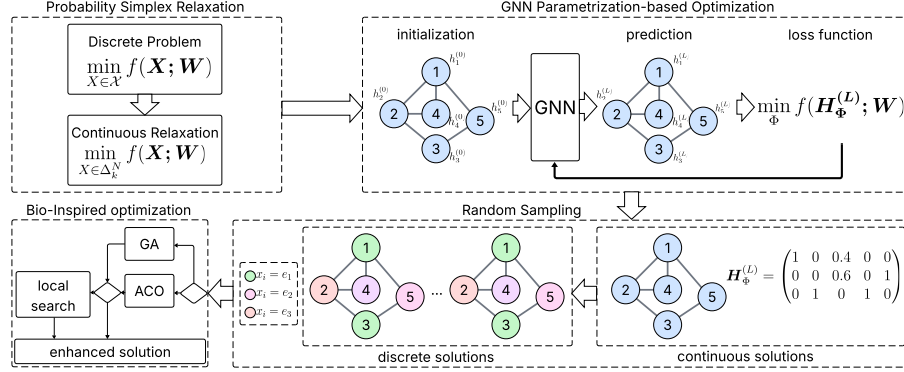


Fig. 1: Scheme of the proposed bio-inspired refinement stage. Adapted from [19].

These research questions motivate the experimental study presented in the remainder of the paper and provide a structured basis for evaluating the effectiveness of the proposed hybrid framework.

The remainder of the paper is organized as follows: the next section introduces the Max- k -Cut optimization problem. Section 3 describes the proposed method. Section 4 and Section 5 present the experimental setup and the numerical results, respectively. Finally, Section 6 outlines directions for future work and concludes the paper.

2 The Max- k -Cut Problem

The Max- k -Cut problem is a classical NP-hard combinatorial optimization problem with applications in, e.g., VLSI design, network reliability, and clustering. Given an undirected weighted graph $G = (V, E)$ with edge weights $w(u, v)$, the goal is to partition the vertices into k disjoint subsets $V_1, V_2, \dots, V_k$ maximizing the total weight of edges crossing between different subsets. The objective value of a k -cut is:

$$cut(V_1, \dots, V_k) = \sum_{\substack{(u,v) \in E \\ u \in V_i, v \in V_j, i \neq j}} w(u, v) \quad (1)$$

with $w(u, v) = 1$ for unweighted graphs. Because exact optimization becomes intractable for large instances, a wide range of metaheuristics has been explored, including Simulated Annealing [18], GAs [11], Harmony Search [12], Tabu Search [13], Scatter Search [17], Estimation of Distribution Algorithms (EDAs) [20], hybrid Particle Swarm Optimization–EDAs [16], and ACO [9]. While promising, these approaches still suffer from the curse of dimensionality, due to the high cost of exploration required by metaheuristic approaches.

Beyond its combinatorial optimization interpretation, Max- k -Cut can also be viewed as a graph pattern partitioning problem. The objective of assigning

vertices to partitions based on edge relationships is closely related to pattern recognition tasks such as graph clustering, community detection, image segmentation, and relational structure analysis. In this context, maximizing inter-partition edge weights corresponds to identifying discriminative structural patterns within graph data.

3 Methods

Relax-Optimize-and-Sample (ROS) [19] is a state-of-the-art GNN-based framework for Max- k -Cut that combines learned graph representations with a continuous relaxation of the discrete assignment, followed by optimization and sampling to recover a discrete partition. We use ROS as a strong initializer and then apply bio-inspired optimization to further improve the cut value.

As for the refinement algorithms, we consider, as a proof-of-concept, GAs and ACO, in different variants. Both algorithms are meant to be post-ROS optimizers: they operate on discrete partitions but can, in principle, be applied to initial solutions produced by other solvers as well. The overall pipeline is summarized in Figure 1.

3.1 Genetic Algorithm

We adopt GAs because they offer a simple and effective balance between exploration and exploitation for discrete combinatorial problems. A candidate solution is encoded as a $|V|$ -dimensional chromosome, where each gene stores the partition label of a vertex. The fitness function is the cut value shown in Eq. (1). All GA variants in this study share the following components.

Initialization. The population is seeded from the ROS solution. One individual is the ROS assignment; the remaining individuals are generated by randomly perturbing a subset of vertices to introduce diversity while preserving structure.

Fitness. Individuals are evaluated by their cut value.

Selection. Parents are chosen via tournament selection.

Crossover. We use uniform crossover: each gene is swapped between parents with a fixed probability.

Mutation. Offspring are mutated by reassigning selected vertices to different partitions with a fixed mutation rate.

Elitism and replacement. Top individuals are preserved across generations, and the remainder of the population is replaced by offspring.

Termination. The algorithm runs for a fixed number of generations and returns the best solution found.

In order to assess different algorithmic configurations and the effect of local search, we evaluate three GA-based variants:

1. GA: standard evolutionary loop using the operators above.
2. GA+LS: a memetic variant where, after each generation, a subset of top individuals is refined by local search. We adopt a first-improvement strategy: vertices are examined for beneficial reassignment moves, accepting the first move that increases the cut, until no improving move remains.
3. MGA+LS: a multi-species GA maintaining multiple interacting subpopulations (species) to promote diversity on a multimodal landscape. Species are initialized by perturbing different subsets of vertices from the ROS solution. Evolutionary operators act within species, while competition and periodic diversity restoration mitigate premature convergence. Elite solutions are periodically refined via local search. The multi-species mechanism promotes the emergence of diverse structural partitioning strategies, similar to adaptive systems in biological pattern recognition.

3.2 Ant Colony Optimization

As is known, ACO works by relying on stigmergic information exchange rather than genetic operators. Its constructive search process naturally matches graph partitioning: each ant incrementally assigns vertices to one of k partitions, guided by pheromone trails and problem-specific heuristics. We initialize ACO using information from ROS to bias early exploration while still allowing the colony to deviate when improvements are found. The distributed decision-making process of the ant colony naturally resembles collective pattern discovery mechanisms observed in biological systems, where global structure emerges through local interactions.

Initialization. The ACO process is initialized by leveraging the output of the ROS model through two complementary mechanisms. The relaxed assignment matrix $X_{\text{relaxed}} \in \mathbb{R}^{k \times n}$ provides soft probabilities for assigning each vertex to one of the k partitions. First, multiple discrete assignments are sampled from the ROS output, and the solution with the best cut value is selected as the initial reference solution. This solution initializes the global best assignment and guides the early stages of the search. Second, pheromone trails are initialized using the relaxed solution as $\tau_{i,p} = \tau_0 + \epsilon X_{\text{relaxed}}(p, i)$, where $\tau_{i,p}$ is the pheromone for assigning vertex i to partition p , τ_0 is a uniform baseline, and ϵ controls the strength of the ROS bias.

Solution construction. Ants combine pheromone information and a greedy heuristic. The pheromone $\tau_{i,p}$ encodes the learned desirability of assigning vertex i to partition p , while the heuristic $h_{i,p}$ estimates the immediate cut gain of the assignment. Each ant samples assignments according to:

$$P_{i,p} = \frac{(\tau_{i,p}^\alpha \cdot h_{i,p}^\beta)^{1/T}}{\sum_{q=1}^k (\tau_{i,q}^\alpha \cdot h_{i,q}^\beta)^{1/T}},$$

where α and β control the relative influence of pheromone and heuristic information, and T is a temperature parameter regulating the exploration–exploitation

trade-off. In addition to probabilistic solution construction, a subset of ants perturbs the current best assignment to intensify the search around promising regions, while others favor either heuristic-driven exploitation or more exploratory behavior to promote diversification. After solutions are constructed, pheromone trails are updated through evaporation and elite-based reinforcement. To prevent premature convergence, adaptive mechanisms adjust parameters such as pheromone influence, heuristic weight, temperature, and evaporation rate based on stagnation indicators (e.g., a number of iterations without improvement). When stagnation persists, stronger perturbations and, if necessary, restart strategies are applied to restore exploration.

In order to assess the effect of local search (similarly to what we did with GA), we evaluate two ACO variants:

1. ACO: constructive ACO process, where solutions are generated through pheromone trails and heuristic guidance, with adaptive diversification mechanisms. No local refinement is applied, allowing this version to serve as a baseline for assessing the standalone contribution of ACO.
2. ACO+LS: the constructive ACO phase is periodically complemented by a local search procedure. At fixed iteration intervals, a subset of high-quality solutions is refined using local optimization operators, including a limited number of 2-opt-style reassignment moves followed by vertex swap operations. Local search is applied before pheromone updates, ensuring that reinforcement is based on improved solution components, while diversification mechanisms (e.g., restarts or perturbations) help mitigate premature convergence.

4 Experimental Setup

We evaluate on instances from the *Gset* benchmark suite [21], a widely used collection for Max-Cut and related problems. Gset contains 67 graphs with varied size and density, including both unweighted and weighted instances. Due to computational budget constraints, we run experiments on 48 instances, selected to cover a range of sizes while keeping runtimes practical. Moreover, we focus on the values of k for which pre-trained weights of ROS are available in the public code repository¹ ($k \in \{2, 3, 10\}$).

ROS Output Analysis To better understand how post-ROS optimization may help, we analyze the confidence of ROS node assignments. For each vertex, we compute the normalized Shannon entropy of the predicted k -way assignment distribution:

$$H = -\frac{\sum_{i=1}^k p_i \log(p_i)}{\log(k)},$$

where p_i is the probability of assigning the vertex to partition i . We consider vertices with $H > 0.3$ *moderately uncertain* and vertices with $H > 0.6$ *highly*

¹ <https://github.com/NetSysOpt/ROS>

uncertain. Across the tested instances and k values, highly uncertain vertices account for roughly 1–2% of all nodes, suggesting that ROS is confident on most assignments while leaving a small set of ambiguous vertices that may benefit from targeted refinement.

Evaluation Protocol For each instance and k , we consider the best cut value obtained by each method under fixed computational budgets (fixed generations/iterations and controlled local-search calls) to ensure fair comparisons. Each algorithm is executed with 5 different seeds, and the results are averaged across runs. We also compute improvements relative to the ROS baseline as a percentage increase in objective value. Hyperparameters are kept fixed to empirically effective settings; the full configuration is reported in the public repository.

Computational Setup All experiments were conducted on a Linux workstation equipped with an Intel Core i9-7940X CPU @ 3.10GHz and 64 GB of RAM. We make our code and hyperparameter configuration publicly available on GitHub².

5 Experimental Results

We compare each variant against its ROS initialization on 48 Gset instances for $k \in \{2, 3, 10\}$. Across all 144 $\langle \text{instance}, k \rangle$ pairs, every bio-inspired method improves the state-of-the-art ROS baseline, with the largest and most consistent gains obtained by variants that include local search. More broadly, these results support the use of bio-inspired metaheuristics as modular post-optimizers that can refine solutions produced by learning-based (and other) initializers. Figure 2 reports the performance profiles, in the form of Dolan-Moré plots [8], over the 48 Gset instances for $k = 2$ and $k = 10$. Complete per-instance results for all tested k values are provided in the public repository.

5.1 Performance Analysis by k -value

$k = 2$ (**Max-2-Cut**). ROS+ACO+LS achieves the best solution on 18 instances (37.5%), demonstrating strong performance particularly on medium-sized graphs. ROS+GA+LS obtains 15 best results (31.3%), showing consistent improvements through local search refinement. ROS+MGA+LS on 14 instances (29.2%) achieves 14 best solutions (29.2%), with particularly strong performance on the largest graphs in the benchmark. The remaining instances show a tie between ROS+ACO and ROS+GA.

$k = 3$ (**Max-3-Cut**). ROS+MGA+LS leads with 16 best solutions (33.3%), demonstrating robust performance across various graph sizes. ROS+GA+LS achieves 15 best results (31.3%). ROS+ACO+LS obtains 10 best solutions (20.8%). ROS+GA achieves 9 best results (18.8%). The remaining best results

² <https://github.com/DIOL-UniTn/bio-enhancing-max-k-cut-solvers.git>

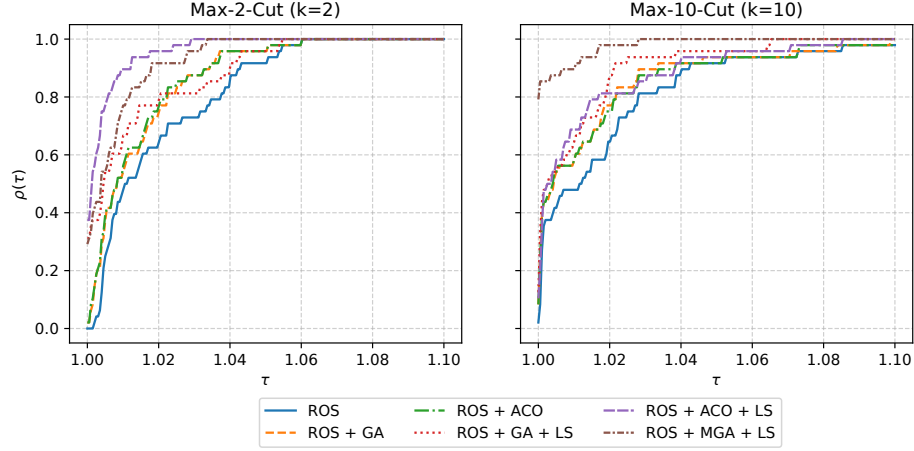


Fig. 2: Performance profile (displayed as Dolan-Moré plots [8], where the performance $\rho(\tau)$ indicates the probability, across all problem instances, that the performance ratio of a given optimizer, w.r.t the performance of the best optimizer for each instance, is within a factor τ of the best possible ratio) over the 48 Gset instances for $k = 2$ (left) and for $k = 10$ (right). Curves toward the top-left part of the plots indicate superior performance.

are split across the non-local-search variants and ties, particularly on the larger graphs G43-G48.

$k = 10$ (Max-10-Cut). ROS+MGA+LS is clearly dominant with 38/48 best solutions (79.2%), while the remaining best solutions are split between ROS+GA+LS (5 instances), ROS+ACO+LS (5 instances), and tied results involving ROS+GA and ROS+ACO. This suggests that as k increases, preserving diversity becomes increasingly important to avoid premature convergence.

5.2 Discussion

To rigorously evaluate the proposed approaches, we analyzed performance across a comprehensive set of 144 experimental scenarios, comprising 48 distinct Gset graphs evaluated across three partition sizes ($k \in \{2, 3, 10\}$). In Table 1, the results summary is reported. The empirical results immediately highlight the critical necessity of local search (LS) for maximizing performance. The standalone evolutionary variants, ROS+GA and ROS+ACO, yielded competitive but modest mean improvements over the baseline ROS of 0.54% and 0.57%, respectively. In contrast, integrating a local search phase enhanced both the consistency and the quality of the final partitions across all algorithmic frameworks.

Among the hybridized methods, ROS+MGA+LS showed the best results, as it returned the highest overall solution in 68 of the 144 instances (47.2%) and

Table 1: Performance comparison of bio-inspired Max- k -Cut solvers. “Best” indicates the number of optimal solutions found per 48-graph subset (with totals occasionally exceeding 48 due to ties), and “Mean Improv.” denotes the average performance gain over the baseline ROS model across all 144 evaluated instances.

Method	Max-2-Cut		Max-3-Cut		Max-10-Cut		Overall
	Best	Rate	Best	Rate	Best	Rate	Mean Improv.
ROS	—	—	—	—	—	—	Baseline
ROS + GA	1	< 2%	9	18.8%	0	< 2%	+0.54%
ROS + ACO	1	< 2%	0	< 2%	0	< 2%	+0.57%
ROS + GA + LS	15	31.3%	15	31.3%	5	10.4%	+0.73%
ROS + ACO + LS	18	37.5%	10	20.8%	5	10.4%	+1.04%
ROS + MGA + LS	14	29.2%	16	33.3%	38	79.2%	+1.24%

achieved a mean improvement of 1.24% over the baseline. Moreover, it proved to yield better outcomes in highly complex search spaces, i.e., when $k = 10$, where ROS+MGA+LS accounted for an impressive 79.2% of the top-performing results. This result can be directly attributed to the multi-species competitive coevolution mechanism, which successfully maintains population diversity and prevents premature convergence—a critical requirement as the number of partitions increases, and the search space expands.

Furthermore, ROS+GA+LS and ROS+ACO+LS variants also demonstrated strong performance profiles. ROS+GA+LS achieved the best known solution in 35 instances (24.3%) with a 0.73% mean improvement over ROS. Its relative strength was most evident in lower-complexity partitioning tasks ($k = 2$ and $k = 3$), where the standard population model converges efficiently without requiring extensive diversity mechanisms. On the other hand, ROS+ACO+LS proved to be more robust, as it returned 33 best solutions (22.9%) and delivered a greater overall mean improvement of 1.04%. The constructive nature of the ACO’s pheromone-guided search resulted in highly balanced performance across all values of k , showing exceptional proficiency specifically for $k = 2$, indicating that the pheromone matrix effectively captures the structural correlations of bipartite divisions.

While all proposed methods successfully improve upon the baseline ROS solutions, the empirical evidence points to a clear scalability trend. As the problem complexity scales with larger values of k , traditional global search mechanisms begin to struggle with the exponentially expanding solution space. The superior performance of ROS+MGA+LS successfully overcomes this bottleneck, demonstrating that aggressive diversity preservation coupled with localized exploitation is the most effective strategy for solving generalized Max- k -Cut problems.

Practical Significance of the Observed Improvements Although the average improvements obtained by the bio-inspired refinement methods are relatively modest (approximately 0.54%-1.24%), several factors support their practical relevance.

First, the improvements are consistently observed across benchmark instances and across multiple refinement strategies. Consistency is particularly important because ROS already represents a strong baseline, making further improvements increasingly difficult to obtain.

Second, in large-scale combinatorial optimization problems, even seemingly small percentage improvements may correspond to substantial absolute gains in the objective value. In applications such as network design, graph clustering, and resource allocation, these gains can translate into meaningful operational benefits.

Finally, the proposed framework is intended as a post-processing step that can be selectively applied when solution quality is prioritized over execution time. Consequently, the additional computational cost may be justified in settings where marginal objective improvements have practical value. Conversely, when computational efficiency is the primary concern, ROS alone may already provide a suitable solution.

6 Conclusions and Future Work

We studied ROS-seeded bio-inspired refinement for Max- k -Cut on a subset of Gset for $k \in \{2, 3, 10\}$. Across all tested $\langle \text{instance}, k \rangle$ pairs, GA- and ACO-based methods improved upon the ROS baseline, and local search consistently increased the gains. The multi-species GA with local search was revealed to be the strongest overall method, particularly for larger k , where diversity preservation becomes critical. While the observed improvements are modest on average, they are consistent and may be valuable in settings where small objective gains justify additional computation. More broadly, this work supports the idea that bio-inspired intelligence can serve as an effective complement to learning-based graph pattern recognition systems. The proposed hybrid framework combines representation learning with adaptive evolutionary refinement, showing how evolutionary and swarm-inspired mechanisms can improve robustness and solution quality on complex graph-structured tasks.

Despite these promising results, our study presents several limitations that also point to directions for future research.

First, the performance of the proposed hybrid algorithms is sensitive to hyperparameter configurations, including population size, mutation rates, pheromone evaporation factors, and local-search frequency. While these parameters were empirically selected based on preliminary pilot runs, optimal configurations may vary across different graph topologies. Future work will therefore investigate systematic hyperparameter tuning.

Second, the integration of local search introduces additional computational overhead, as repeated fitness evaluations become expensive on large-scale graphs. In this study, we mitigate this issue by restricting local search to high-quality candidate solutions and enforcing bounded iteration budgets. Future research could explore more efficient local search strategies or adaptive mechanisms that dynamically control their application.

Finally, due to computational constraints, our experimental evaluation is limited to a subset of the Gset benchmark and does not include all algorithmic variants on the largest instances. Extending the evaluation to the full benchmark suite, as well as to additional synthetic and real-world graphs, will be important to further assess scalability and generalization.

Beyond these aspects, promising directions include the exploration of alternative bio-inspired optimizers and novel hybridization strategies, as well as the investigation of different initialization schemes beyond ROS-based seeding to improve early-stage diversity and avoid premature convergence.

References

1. Bairamkulov, R., Friedman, E.: Graphs in VLSI circuits and systems. In: *Graphs in VLSI*, pp. 59–100. Springer (2022)
2. Bucur, D., Iacca, G.: Influence maximization in social networks with genetic algorithms. In: *European Conference on the Applications of Evolutionary Computation*. pp. 379–392. Springer International Publishing (2016)
3. Bucur, D., Iacca, G., Marcelli, A., Squillero, G., Tonda, A.: Multi-Objective Evolutionary Algorithms for Influence Maximization in Social Networks. In: *European Conference on the Applications of Evolutionary Computation*. pp. 221–233. Springer, Cham (2017)
4. Corso, G., Stark, H., Jegelka, S., Jaakkola, T., Barzilay, R.: Graph neural networks. *Nature Reviews Methods Primers* **4**(1), 17 (2024)
5. Cunegatti, E., Bucur, D., Iacca, G.: Understanding Sparse Neural Networks from their Topology via Multipartite Graph Representations. *Transactions on Machine Learning Research* (2024)
6. Cunegatti, E., Custode, L.L., Iacca, G.: Many-Objective Evolutionary Influence Maximization: Balancing Spread, Budget, Fairness, and Time. In: *Genetic and Evolutionary Computation Conference Companion*. pp. 655–658 (2024)
7. Cunegatti, E., Iacca, G., Bucur, D.: Large-scale multi-objective influence maximization with network downscaling. In: *Parallel Problem Solving from Nature*. vol. 13399 (2022)
8. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. *Mathematical programming* **91**(2), 201–213 (2002)
9. Gao, L., Zeng, Y., Dong, A.: An ant colony algorithm for solving Max-cut problem. *Progress in Natural Science* **18**(9), 1173–1178 (2008)
10. Genetti, S., Ribaga, E., Cunegatti, E., Lotito, Q.F., Iacca, G.: Influence maximization in hypergraphs using multi-objective evolutionary algorithms. In: *Parallel Problem Solving from Nature*, pp. 217–235. Springer Nature Switzerland (2024)
11. Kim, Y.H., Geem, Z.W., Yoon, Y.: Population-based redundancy control in genetic algorithms: Enhancing max-cut optimization. *Mathematics* **13**(9), 1409 (2025)
12. Kim, Y.H., Yoon, Y., Geem, Z.W.: A comparison study of harmony search and genetic algorithm for the max-cut problem. *Swarm and evolutionary computation* **44**, 130–135 (2019)
13. Kochenberger, Gary A and Hao, Jin-Kao and Lü, Zhipeng and Wang, Haibo and Glover, Fred: Solving large scale max cut problems via tabu search. *Journal of Heuristics* **19**(4), 565–571 (2013)

14. Konotopska, K., Iacca, G.: Graph-Aware Evolutionary Algorithms for Influence Maximization. In: Genetic and Evolutionary Computation Conference Companion. pp. 1467–1475 (2021)
15. Li, R., Yuan, X., Radfar, M., Marendy, P., Ni, W., O’Brien, T.J., Casillas-Espinosa, P.M.: Graph signal processing, graph neural network and graph learning on biological data: a systematic review. *IEEE Reviews in Biomedical Engineering* **16**, 109–135 (2021)
16. Lin, G., Guan, J.: An Integrated Method Based on PSO and EDA for the Max-Cut Problem. *Computational intelligence and neuroscience* **2016**(1), 3420671 (2016)
17. Martí, Rafael and Duarte, Abraham and Laguna, Manuel: Advanced scatter search for the max-cut problem. *INFORMS Journal on Computing* **21**(1), 26–38 (2009)
18. Myklebust, T.G.: Solving maximum cut problems by simulated annealing (2015), arXiv preprint arXiv:1505.03068
19. Qiu, Y., Xue, Y., Wang, A., Wang, Y., Shi, Q., Luo, Z.Q.: ROS: A GNN-based Relax-Optimize-and-Sample Framework for Max-k-Cut Problems (2024), arXiv preprint arXiv:2412.05146
20. de Sousa, S., Haxhimusa, Y., Kropatsch, W.G.: Estimation of distribution algorithm for the max-cut problem. In: International Workshop on Graph-Based Representations in Pattern Recognition. pp. 244–253. Springer, Berlin, Heidelberg (2013)
21. Ye, Y.: The G-set Dataset (2003), <https://web.stanford.edu/~yyye/yyye/Gset/>, last accessed: 18 June 2026
22. Zhang, H., Nie, F., Li, X.: Large-scale clustering with structured optimal bipartite graph. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(8), 9950–9963 (2023)