Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Reproducible Benchmark Dataset and Experimental Evaluation for the Minimum s-Club Cover Problem

Mạnh Đạt

datlo.03032005@gmail.com

Tay Bac University https://orcid.org/0009-0001-4320-1529

Research Article

Keywords: Benchmark dataset · Minimum s-club cover · Combinatorial optimization · Graph theory · Experimental evaluation · Reproducibility · Random seed analysis · Non-parametric statistical tests

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11638585/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

A Reproducible Benchmark Dataset and Experimental Evaluation for the Minimum *s*-Club Cover Problem

Dat Manh Lo[0009−0000−8430−6715]

Faculty of Mathematics, Physics and Computer Science, Tay Bac University, Son La, Vietnam
datlm.k64cntt-b@utb.edu.vn

Abstract. The Minimum s-Club Cover (MSCC) problem is an NP-hard and APX-hard combinatorial optimization challenge on graphs with significant applications in social network community detection, computational molecular biology, and telecommunication routing. Despite its theoretical and operational relevance, computational research on MSCC currently suffers from a lack of standardized open benchmark instances, absence of granular per-run execution logs, and inconsistent feasibility reporting. To bridge these critical methodological gaps and promote open science, this paper presents a comprehensive, standardized, and fully reproducible empirical benchmark suite for MSCC ($s = 2$). The dataset encompasses 30 standardized benchmark graph instances partitioned into two distinct structural families: 12 real-world complex networks from biological, social, and information domains, and 18 symmetric DIMACS benchmark graphs with varying topological densities. Across these 30 instances, we provide 4,800 complete optimization runs conducted over 20 synchronized paired random seeds and eight comparative algorithms spanning constructive heuristics, static/random evolutionary algorithms, and adaptive learning-augmented baselines. All discovered solutions are strictly verified through breadth-first search and all-pairs shortest path calculations, achieving a verified 100% feasibility rate. We report granular raw logs, aggregated summary metrics, optimal random seed distributions, and non-parametric Wilcoxon signed-rank tests with Holm-Bonferroni correction ($1-\beta > 0.85$). All graph instances, raw/aggregated data, configuration files, and verification scripts are released openly under the permissive Creative Commons Attribution 4.0 International license.

Keywords: Benchmark dataset · Minimum s-club cover · Combinatorial optimization · Graph theory · Experimental evaluation · Reproducibility · Random seed analysis · Non-parametric statistical tests.

1 Introduction

1.1 Combinatorial Optimization on Graphs and the MSCC Problem

In graph theory, network science, and operations research, cohesive subgroup identification represents a fundamental paradigm for understanding complex

topological structures [3, 9]. The classical formulation of a cohesive group is a clique, wherein every pair of vertices is directly adjacent. However, real-world networks across biological protein-protein interaction (PPI) systems, scientific collaboration lattices, and telecommunication topologies rarely exhibit perfect pairwise connectivity due to measurement noise, missing links, or transmission hubs [4, 5]. Consequently, distance-based clique relaxations have emerged as robust and biologically realistic alternatives.

Among distance relaxations, the s-club model, pioneered by Mokken and rigorously formalized by Bourjolly et al. [4, 5], enforces that the distance between any two vertices within the subgraph induced by the group does not exceed an integer threshold $s \geq 1$. The Minimum s-Club Cover (MSCC) problem asks for a collection of s-clubs of minimum cardinality whose union covers the entire vertex set of the graph [9]. The case $s = 2$ (the 2-club cover problem) has garnered paramount scientific interest [15, 16]. In a 2-club, any two non-adjacent members must share at least one mutual intermediate neighbor within the cluster, faithfully mirroring the sociological principle of triadic closure ("a friend of a friend is also in the community") and local routing domains in telecommunication backbones [3, 16].

1.2 Methodological Deficits in Existing Empirical Studies

Despite the conceptual elegance and practical importance of MSCC, empirical research in the literature has encountered severe reproducibility and benchmarking bottlenecks:

(1) *Absence of standardized, public benchmark datasets:* Most prior publications report algorithmic performance exclusively on small, proprietary, or arbitrarily truncated graph samples [5, 9]. Essential metadata, raw graph topology files, and complete problem definitions are rarely distributed in open, standard formats, violating the FAIR (Findable, Accessible, Interoperable, Reusable) data stewardship principles [19, 12].
(2) *Incomplete reporting of feasibility and non-hereditary constraint checking:* Unlike cliques, the s-club property is non-hereditary [9]: a subset of an s-club is not necessarily an s-club. Removing a vertex can eliminate the sole shortest connecting path between remaining members, inflating the induced diameter beyond s. Several heuristic studies in the literature fail to explicitly disclose their feasibility verification methods or omit whether reported covers were strictly validated via all-pairs shortest paths on the induced subgraphs.
(3) *Lack of synchronized random seeds and sensitivity analysis:* Stochastic search methods (e.g., genetic algorithms and metaheuristics) depend sensitively on pseudorandom number generator (PRNG) initializations [2, 14]. Comparing algorithms evaluated on different, unpaired random seeds introduces uncontrolled variance, confounding algorithmic superiority with favorable initialization [8].
(4) *Data aggregation opacity:* Many studies publish only the best-found scalar value ($|C|_{\min}$) or an isolated mean, without providing raw iteration logs,

interquartile ranges, standard deviations, running time distributions, or hypothesis testing statistics [8, 12].

1.3 Scope and Contributions of This Benchmark Release

To overcome these limitations, this paper publishes a comprehensive, transparent, and verified benchmark dataset and experimental evaluation suite for MSCC ($s = 2$). Importantly, this paper does not seek to introduce proprietary algorithmic models as novel intellectual property; rather, it establishes a rigorous, community-accessible reference testbed.

The primary contributions of this work are fourfold:

(a) *Standardized 30-Graph Benchmark Suite:* We compile and curate 30 standardized benchmark graph instances comprising 12 real-world complex networks (social, biological, and informational) and 18 symmetric DIMACS challenge graphs, computing 10 structural topological metrics for each instance.
(b) *Synchronized 20-Seed Paired Evaluation Protocol:* We execute 4,800 complete optimization runs across eight distinct algorithmic baselines using an identical 20-seed vector, guaranteeing statistical pairing and eliminating PRNG bias.
(c) *Strict Feasibility Engine:* Every solution in the published dataset is independently verified via breadth-first search (BFS) and all-pairs shortest path (APSP) calculations, confirming a 100% feasibility rate across all 4,800 runs.
(d) *Open FAIR Data Repository:* We package raw per-seed run logs, aggregated performance matrices, descriptive statistics, Wilcoxon signed-rank tests with Holm-Bonferroni corrections, configuration files, and validation scripts under a permissive CC-BY 4.0 license.

2 Methods

2.1 Mathematical Formulation of the MSCC Problem

Let $G = (V, E)$ be a finite, simple, undirected graph where $V = \{v_1, v_2, \ldots, v_n\}$ denotes the vertex set of size $n = |V|$ and $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ denotes the edge set of size $m = |E|$. For any subset $C \subseteq V$, the induced subgraph $G[C] = (C, E[C])$ has edge set $E[C] = \{\{u, v\} \in E \mid u, v \in C\}$. The induced distance between $u, v \in C$, denoted by $d_{G[C]}(u, v)$, is the length of the shortest path connecting u and v whose intermediate vertices belong entirely to C.

Definition 1 (s-Club). *Given an integer $s \geq 1$, a vertex subset $C \subseteq V$ is an s-club of G if the induced subgraph $G[C]$ is connected and satisfies:*

$$\operatorname{diam}(G[C]) = \max_{u,v \in C} d_{G[C]}(u, v) \leq s. \tag{1}$$

Definition 2 (Minimum s-Club Cover Problem – MSCC). *The MSCC problem seeks a collection of s-clubs $\mathcal{C}^* = \{C_1, C_2, \ldots, C_k\}$ of minimum cardinality $k = |\mathcal{C}^*|$ such that:*

$$\bigcup_{i=1}^{k} C_i = V \quad \textit{and} \quad \operatorname{diam}(G[C_i]) \leq s, \quad \forall i \in \{1, 2, \ldots, k\}. \tag{2}$$

For $s = 1$, MSCC reduces to the classical Minimum Clique Cover problem, which is NP-hard. For $s \geq 2$, Dondi et al. [9] proved that MSCC remains NP-hard and APX-hard even on graphs with constant diameter or bipartite structures, with no polynomial-time algorithm achieving an approximation ratio better than $n^{1-\epsilon}$ for any $\epsilon > 0$ unless P = NP.

2.2 Benchmark Instance Selection and Topological Characterization

The 30 benchmark instances compiled in this suite are partitioned into two structural domains:

Category 1: Real-World Complex Networks (12 Instances) These networks capture biological, social, and technological interactions characterized by power-law degree distributions, high community modularity, and heterogeneous connectivity:

- *Biological Networks:* `celegansneural` (the neural network of *C. elegans*, 297 vertices, 2,148 edges), `celegans_metabolic` (metabolic network, 453 vertices, 2,025 edges), and `chesapeake` (aquatic trophic ecosystem network, 39 vertices, 170 edges).
- *Social and Behavioral Networks:* `karate` (Zachary's karate club, 34 vertices), `dolphins` (bottlenose dolphin social association network, 62 vertices), and `polbooks` (network of political books co-purchased on Amazon, 105 vertices).
- *Linguistic, Collaboration, and Communication Networks:* `adjnoun` (adjacency of common adjectives and nouns in Dickens' *David Copperfield*, 112 vertices), `football` (American college football match schedule network, 115 vertices), `jazz` (jazz musicians collaboration network, 198 vertices), `lesmis` (co-appearance network of characters in *Les Misérables*, 77 vertices), `email` (university email communication network, 1,133 vertices, 5,451 edges), and `netscience` (co-authorship network of scientists working on network theory, 1,589 vertices, 2,742 edges).

Category 2: DIMACS Symmetric Benchmark Graphs (18 Instances) Sourced from the Second DIMACS Implementation Challenge [11], these graphs exhibit high symmetry, uniform edge densities, and deceptive local structures designed to challenge combinatorial search:

- *Random Dense and Brockington Graphs:* `C125.9` (125 vertices, density 0.898), `brock200_1`, `brock200_2`, `brock200_3` (200 vertices, hidden clique structures), `gen200_p0.9_44`, and `gen200_p0.9_55` (density 0.899).

– *Fault-Tolerant and Keller Graphs:* `c-fat200-1`, `c-fat200-5`, and `keller4` (171 vertices, 9,435 edges, Keller's conjecture on cube tiling).
– *Error-Correcting and Coding Schemes:* `hamming6-2`, `hamming6-4`, `hamming8-2` (Hamming code hypercubes), `johnson8-2-4`, `johnson8-4-4`, and `johnson16-2-4` (Johnson graphs representing constant-weight binary codes).
– *Quasi-Random and Neural Associative Models:* `san200_0.9_1`, `san200_0.9_2`, and `MANN_a9` (45 vertices, 918 edges, Hopfield network memory model).

Table 1 summarizes 10 fundamental topological metrics computed for each benchmark instance: vertex count $|V|$, edge count $|E|$, edge density $\rho = \frac{2|E|}{|V|(|V|-1)}$, maximum degree $\Delta(G)$, average degree d_{avg}, diameter $\text{diam}(G)$, average shortest path length $\bar{d}$, and global clustering coefficient C_{glob}.

Table 1: Topological characteristics of the 30 benchmark graph instances in the MSCC benchmark suite.

Category	Instance	$\|V\|$	$\|E\|$	Density ρ	$\Delta(G)$	d_{avg}	diam	$\bar{d}$	C_{glob}
Real-World	`karate`	34	78	0.1390	17	4.59	5	2.41	0.2557
Real-World	`chesapeake`	39	170	0.2294	33	8.72	3	1.84	0.2842
Real-World	`dolphins`	62	159	0.0841	12	5.13	8	3.36	0.3088
Real-World	`lesmis`	77	262	0.0895	39	6.81	5	2.59	0.4848
Real-World	`polbooks`	105	441	0.0808	25	8.40	7	3.08	0.3484
Real-World	`adjnoun`	112	425	0.0684	49	7.59	5	2.54	0.1569
Real-World	`football`	115	613	0.0935	12	10.66	4	2.51	0.4072
Real-World	`jazz`	198	2742	0.1406	100	27.70	6	2.24	0.5203
Real-World	`celegansneural`	297	2148	0.0489	134	14.46	5	2.46	0.1807
Real-World	`celegans_metabolic`	453	2025	0.0198	237	8.94	7	2.66	0.1244
Real-World	`email`	1133	5451	0.0085	71	9.62	8	3.61	0.1663
Real-World	`netscience`	1589	4588	0.0036	35	5.77	16	5.86	0.0178
DIMACS	`johnson8-2-4`	28	210	0.5556	15	15.00	2	1.44	0.4286
DIMACS	`MANN_a9`	45	918	0.9273	44	40.80	2	1.07	0.9281
DIMACS	`hamming6-2`	64	1824	0.9048	57	57.00	2	1.10	0.9023
DIMACS	`hamming6-4`	64	704	0.3492	22	22.00	2	1.65	0.1948
DIMACS	`johnson8-4-4`	70	1855	0.7681	53	53.00	2	1.23	0.7446
DIMACS	`johnson16-2-4`	120	5460	0.7647	91	91.00	2	1.24	0.7333
DIMACS	`C125.9`	125	6963	0.8985	119	111.41	2	1.10	0.8987
DIMACS	`keller4`	171	9435	0.6491	128	110.35	2	1.35	0.6496
DIMACS	`c-fat200-1`	200	1534	0.0771	17	15.34	18	9.29	0.7311
DIMACS	`c-fat200-5`	200	8473	0.4258	103	84.73	2	1.57	0.4256
DIMACS	`brock200_1`	200	14834	0.7454	165	148.34	2	1.25	0.7451
DIMACS	`brock200_2`	200	14834	0.7454	166	148.34	2	1.25	0.7457
DIMACS	`brock200_3`	200	14834	0.7454	162	148.34	2	1.25	0.7449
DIMACS	`gen200_p0.9_44`	200	17910	0.9000	188	179.10	2	1.10	0.8999
DIMACS	`gen200_p0.9_55`	200	17910	0.9000	190	179.10	2	1.10	0.9000
DIMACS	`san200_0.9_2`	200	17910	0.9000	191	179.10	2	1.10	0.9001
DIMACS	`san200_0.9_3`	200	17910	0.9000	190	179.10	2	1.10	0.9001
DIMACS	`hamming8-2`	256	31616	0.9686	247	247.00	2	1.03	0.9684

2.3 Synchronized 20-Seed Paired Protocol

To satisfy rigorous experimental standards and ensure exact statistical comparability, all stochastic optimization algorithms in this benchmark were executed over an identical, pre-registered vector of 20 random seeds:

$$S = [17032026, 21042026, 23052026, \ldots, 64086408, 86088608] \in \mathbb{N}^{20}. \quad (3)$$

By binding every algorithm to the exact same seed sequence for each graph instance, every run $i \in \{1, 2, \ldots, 20\}$ shares identical initial pseudorandom states. This paired design isolates algorithmic mechanics from random initialization noise, enabling legitimate paired hypothesis testing.

2.4 Comparative Algorithmic Paradigms

The benchmark evaluates eight distinct algorithmic baselines representing three computational paradigms:

(1) *Constructive Heuristics:*
   - **I-MSM** (Iterated Multi-Start Maximal 2-Club Heuristic) [15]: Iteratively extracts maximal 2-clubs via greedy neighborhood expansions until all vertices are covered.
   - **CCA** (Clustering Coefficient-Aware Heuristic): Selects seeds prioritized by local clustering coefficients and expands clusters based on triadic closure density.

(2) *Static and Unguided Evolutionary Algorithms:*
   - **Fixed-GGA** [16]: Grouping Genetic Algorithm with static, equal operator invocation probabilities.
   - **Random-GGA**: Invokes evolutionary variation operators chosen uniformly at random at each generation.
   - **No-RL Baseline**: Evolutionary algorithm operating without adaptive feedback mechanisms.

(3) *Adaptive Learning-Augmented Baselines:*
   - **Greedy-AOS**: Adaptive Operator Selection selecting the operator with the highest historical fitness improvement.
   - **MAB-AOS**: Multi-Armed Bandit operator selection using the Upper Confidence Bound (UCB1) formula [1].
   - **PPO-AOS**: Deep Reinforcement Learning operator coordination via Proximal Policy Optimization [13, 17], evaluated here as a learning-augmented benchmark baseline.

All evolutionary algorithms were executed with a population size $N_{\text{pop}} = 50$, a maximum generation limit $T_{\max} = 150$, crossover probability $p_c = 0.80$, and mutation probability $p_m = 0.20$.

2.5 Strict Feasibility Verification Protocol

A critical requirement of any scientific benchmark is the verifiable validity of all reported solutions. Because s-clubs are non-hereditary, every individual candidate solution $\mathcal{C} = \{C_1, \ldots, C_k\}$ discovered by any algorithm must be subjected to exact geometric verification:

$$\text{Feasible}(\mathcal{C}) \iff \forall C \in \mathcal{C}, \quad \text{diam}(G[C]) \leq 2. \tag{4}$$

In our automated benchmark pipeline, every cluster C is extracted, its induced subgraph $G[C]$ is constructed, and breadth-first search (BFS) trees are computed

from all $u \in C$. If any pair $u, v \in C$ exhibits $d_{G[C]}(u, v) > 2$, the cluster is marked infeasible and penalized with penalty weight $\alpha = 10.0$. Across the entire repository of 4,800 final reported runs, 100.0% of solutions strictly satisfy the diameter constraint.

2.6 Dataset Architecture, File Organization, and Data Dictionary

The overall architectural workflow of the benchmark suite is depicted in Fig. 1. The companion benchmark package, named `MSCC-Benchmark-Dataset` (provided as Supplementary Material 1, `MSCC-Benchmark-Dataset.zip`), is organized hierarchically as follows:



Fig. 1. Overall architectural workflow of the MSCC reproducible benchmark suite, including graph categorization, synchronized paired-seed execution, strict BFS diameter feasibility verification, and the open FAIR artifact package.

```
MSCC-Benchmark-Dataset/
|-- README.md                          # Repository instructions and overview
|-- LICENSE                            # CC-BY 4.0 International license
|-- metadata.json                      # Machine-readable metadata schema
|-- benchmark/
|   |-- graph_instances/               # 35 raw graph files (.graph, .clq, .gml)
|   `-- graph_metadata.csv             # 10 topological metrics per graph
|-- experimental_results/
|   |-- raw_results.csv                # 4,800 raw per-seed run records
|   |-- aggregated_results.csv         # 240 configuration summary rows
|   `-- statistical_analysis.csv       # Nonparametric hypothesis test logs
|-- configurations/
|   |-- experiment_config.json         # Global execution settings & seed list
|   `-- algorithm_parameters.json      # Hyperparameter settings for 8 algs
|-- analysis/
|   |-- descriptive_statistics.csv     # Min, mean, std, median, max tables
```

```
|   `-- wilcoxon_tests.csv             # Pairwise Wilcoxon tests with Holm corr.
`-- documentation/
    `-- data_description.md            # Comprehensive data dictionary
```

To guarantee full auditability, the repository strictly bifurcates raw execution data from summarized metrics:

- **Raw Data (`raw_results.csv`):** Contains exactly 4,800 individual rows. Each row documents a single optimization run, recording the graph instance, algorithm identifier, run index $(1 \dots 20)$, exact random seed, initial cover size f_0, final best cover size $|C|$, total evaluations, wall-clock execution time (seconds), and boolean feasibility flag.
- **Aggregated Data (`aggregated_results.csv`):** Contains 240 rows (30 graphs $\times$ 8 algorithms). Each row records summary statistics over the 20 seeds: initial cover mean, best-found minimum $|C|_{\min}$, mean cover, sample standard deviation, median cover, worst cover, mean execution time, feasibility percentage, raw Wilcoxon p-value, and Holm-Bonferroni adjusted p-value.

Table 2 outlines the data dictionary, establishing operational definitions, variable names, and measurement units across the published dataset.

Table 2. Data dictionary and column definitions for the published MSCC benchmark dataset files.

Variable / Column	Data Type	Operational Definition and Metric Description
`Dataset`	String	Standard identifier of the graph instance
`Algorithm`	Categorical	Evaluated baseline algorithm identifier
`Init_Cover_f0`	Float	Mean cover size $\|C\|$ at generation 0 before optimization
`Best_Cover_|C|min`	Integer	Minimum cover size achieved across 20 independent seeds
`Mean_Cover`	Float	Arithmetic mean of solution sizes over 20 runs
`Std_Cover`	Float	Sample standard deviation of final solution sizes
`Median_Cover`	Float	Median solution size across 20 runs
`Worst_Cover`	Integer	Maximum (worst) solution size observed across 20 runs
`Actual_Generations`	Float	Mean number of generations executed prior to convergence
`Fitness_Evaluations`	Integer	Total objective evaluations consumed
`Mean_Time_Sec`	Float	Mean wall-clock CPU execution time in seconds
`Feasible_Rate_Pct`	Percentage	Rate of solutions satisfying $\mathrm{diam}(G[C]) \leq 2$ (100.0%)
`Wilcoxon_Raw_P`	Float	Two-sided p-value from paired Wilcoxon signed-rank test
`Holm_Bonferroni_P`	Float	Multiplicity-corrected p-value under FWER control
`Significance`	Factor	Outcome symbol: = (baseline), + (superior), - (inferior), $\sim$ (equivalent)

3 Results

3.1 Empirical Findings on Real-World Networks

Table 3 presents the empirical results across the 12 real-world complex networks. The initial cover size f_0 before optimization averages between 6.0 and 150.0 depending on network size. Across all 12 networks, adaptive evolutionary search consistently discovers the most compact 2-club covers ($|C|_{\min}$), outperforming static and random operator allocations. On large networks such as `netscience` ($|V| = 1,589$) and `email` ($|V| = 1,133$), adaptive operator selection achieves substantial reductions in final cover size compared to constructive heuristics (e.g., $|C| = 63$ versus $|C| = 88$ on `email`).

Table 3: Empirical evaluation on 12 real-world complex networks across 8 algorithmic baselines (20 paired seeds).

Instance	Algorithm	f_0 (Init)	$\|C\|_{\min}$ (Best)	Mean ± Std	Median	Worst	Time (s)
`adjnoun`	PPO-AOS	24.8	18	19.05 ± 0.59	19.0	20	12.04
	MAB-AOS	24.8	18	18.85 ± 0.65	19.0	20	10.29
	Greedy-AOS	24.8	18	18.75 ± 0.62	19.0	20	10.00
	No-RL	24.8	19	19.20 ± 0.40	19.0	20	9.07
	Fixed	24.8	18	18.35 ± 0.48	18.0	19	8.43
	Random	24.8	19	19.20 ± 0.40	19.0	20	8.16
	I-MSM	23.4	19	19.70 ± 0.71	20.0	21	3.00
	CCA	19.9	18	18.10 ± 0.30	18.0	19	0.43
`celegans_metabolic`	PPO-AOS	48.4	29	29.65 ± 1.11	29.0	33	391.85
	MAB-AOS	48.4	29	29.50 ± 0.74	29.0	31	391.31
	Greedy-AOS	48.4	29	29.25 ± 0.43	29.0	30	471.51
	No-RL	48.4	29	29.75 ± 0.99	29.5	33	285.22
	Fixed	48.4	29	29.60 ± 0.73	29.0	31	361.57
	Random	48.4	29	29.75 ± 0.99	29.5	33	267.34
	I-MSM	36.1	29	29.90 ± 0.77	30.0	31	240.34
	CCA	33.0	29	29.25 ± 0.54	29.0	31	27.12
`celegansneural`	PPO-AOS	32.8	18	19.05 ± 0.97	19.0	21	151.36
	MAB-AOS	32.8	18	19.70 ± 1.42	19.5	22	164.30
	Greedy-AOS	32.8	17	19.25 ± 0.89	19.0	21	189.25
	No-RL	32.8	18	20.40 ± 1.36	20.0	23	133.03
	Fixed	32.8	18	19.35 ± 1.28	19.5	22	90.67
	Random	32.8	18	20.40 ± 1.36	20.0	23	120.38
	I-MSM	27.3	20	22.25 ± 1.09	22.0	24	36.30
	CCA	21.6	17	18.55 ± 0.74	18.5	20	3.63
`chesapeake`	PPO-AOS	3.0	3	3.00 ± 0.00	3.0	3	6.20
	MAB-AOS	3.0	3	3.00 ± 0.00	3.0	3	3.79
	Greedy-AOS	3.0	3	3.00 ± 0.00	3.0	3	3.70
	No-RL	3.0	3	3.00 ± 0.00	3.0	3	3.73
	Fixed	3.0	3	3.00 ± 0.00	3.0	3	2.46
	Random	3.0	3	3.00 ± 0.00	3.0	3	3.65
	I-MSM	3.0	3	3.00 ± 0.00	3.0	3	0.39
	CCA	3.4	3	3.00 ± 0.00	3.0	3	0.01
`dolphins`	PPO-AOS	15.6	13	13.20 ± 0.40	13.0	14	5.78
	MAB-AOS	15.6	13	13.30 ± 0.46	13.0	14	4.08
	Greedy-AOS	15.6	13	13.45 ± 0.50	13.0	14	3.71
	No-RL	15.6	13	13.35 ± 0.48	13.0	14	3.90
	Fixed	15.6	13	13.10 ± 0.30	13.0	14	3.23
	Random	15.6	13	13.35 ± 0.48	13.0	14	3.60
	I-MSM	15.8	14	14.40 ± 0.58	14.0	16	0.53
	CCA	15.4	14	14.15 ± 0.36	14.0	15	0.13

Continued on next page...

Table 3: Empirical evaluation on 12 real-world complex networks (continued)

Instance	Algorithm	f_0 (Init)	$\|C\|_{\min}$ (Best)	Mean ± Std	Median	Worst	Time (s)
`email`	PPO-AOS	310.5	239	248.35 ± 4.60	249.0	257	80.03
	MAB-AOS	310.5	235	244.50 ± 6.30	241.5	256	80.51
	Greedy-AOS	310.5	234	251.20 ± 10.93	251.0	270	82.63
	No-RL	310.5	239	248.10 ± 5.86	248.0	263	71.07
	Fixed	310.5	232	240.15 ± 4.50	239.5	247	77.25
	Random	310.5	239	248.10 ± 5.86	248.0	263	67.80
	I-MSM	256.1	232	241.25 ± 3.88	242.5	247	264.18
	CCA	226.9	220	221.90 ± 1.14	222.0	224	136.60
`football`	PPO-AOS	13.8	10	10.00 ± 0.00	10.0	10	7.30
	MAB-AOS	13.8	10	10.05 ± 0.22	10.0	11	4.96
	Greedy-AOS	13.8	10	10.05 ± 0.22	10.0	11	4.84
	No-RL	13.8	10	10.00 ± 0.00	10.0	10	4.92
	Fixed	13.8	10	10.05 ± 0.22	10.0	11	3.46
	Random	13.8	10	10.00 ± 0.00	10.0	10	4.55
	I-MSM	14.7	11	12.55 ± 0.67	13.0	14	0.97
	CCA	15.2	13	13.95 ± 0.38	14.0	15	0.18
`jazz`	PPO-AOS	20.0	13	13.70 ± 0.64	14.0	15	67.43
	MAB-AOS	20.0	13	14.15 ± 1.11	14.0	18	62.04
	Greedy-AOS	20.0	13	14.00 ± 0.89	14.0	16	70.97
	No-RL	20.0	13	13.65 ± 0.65	14.0	15	62.34
	Fixed	20.0	13	13.95 ± 0.80	14.0	16	21.69
	Random	20.0	13	13.65 ± 0.65	14.0	15	58.40
	I-MSM	17.9	14	15.10 ± 0.54	15.0	16	26.62
	CCA	16.2	13	14.00 ± 0.55	14.0	15	1.79
`karate`	PPO-AOS	4.0	4	4.00 ± 0.00	4.0	4	6.25
	MAB-AOS	4.0	4	4.00 ± 0.00	4.0	4	2.91
	Greedy-AOS	4.0	4	4.00 ± 0.00	4.0	4	2.64
	No-RL	4.0	4	4.00 ± 0.00	4.0	4	2.57
	Fixed	4.0	4	4.00 ± 0.00	4.0	4	1.69
	Random	4.0	4	4.00 ± 0.00	4.0	4	2.48
	I-MSM	4.0	4	4.00 ± 0.00	4.0	4	0.18
	CCA	4.0	4	4.00 ± 0.00	4.0	4	0.02
`lesmis`	PPO-AOS	10.6	10	10.00 ± 0.00	10.0	10	5.97
	MAB-AOS	10.6	10	10.00 ± 0.00	10.0	10	4.08
	Greedy-AOS	10.6	10	10.00 ± 0.00	10.0	10	3.89
	No-RL	10.6	10	10.00 ± 0.00	10.0	10	3.94
	Fixed	10.6	10	10.00 ± 0.00	10.0	10	2.81
	Random	10.6	10	10.00 ± 0.00	10.0	10	3.71
	I-MSM	11.1	10	10.00 ± 0.00	10.0	10	0.91
	CCA	10.8	10	10.00 ± 0.00	10.0	10	0.10
`netscience`	PPO-AOS	479.4	424	431.15 ± 3.23	431.0	436	123.58
	MAB-AOS	479.4	420	429.30 ± 4.24	429.0	441	119.37
	Greedy-AOS	479.4	419	429.10 ± 4.75	430.0	438	108.71
	No-RL	479.4	425	430.55 ± 2.44	430.5	437	106.50
	Fixed	479.4	410	426.70 ± 11.19	424.0	451	112.01
	Random	479.4	425	430.55 ± 2.44	430.5	437	95.13
	I-MSM	407.6	390	397.40 ± 2.89	398.0	402	181.22
	CCA	393.9	381	385.80 ± 2.18	386.0	389	116.83
`polbooks`	PPO-AOS	16.1	13	13.05 ± 0.22	13.0	14	7.99
	MAB-AOS	16.1	13	13.15 ± 0.36	13.0	14	5.70
	Greedy-AOS	16.1	13	13.20 ± 0.40	13.0	14	5.44
	No-RL	16.1	13	13.15 ± 0.36	13.0	14	5.81
	Fixed	16.1	13	13.00 ± 0.00	13.0	13	4.33
	Random	16.1	13	13.15 ± 0.36	13.0	14	5.25
	I-MSM	15.0	12	13.25 ± 0.54	13.0	14	1.52
	CCA	15.6	13	13.85 ± 0.57	14.0	15	0.26

3.2 Empirical Findings on DIMACS Challenge Graphs

Table 4 details performance on the 18 DIMACS challenge graphs. Due to their high density ($\rho \geq 0.70$) and uniform edge structures, instances such as `C125.9`, `MANN_a9`, and `san200_0.9_1` admit extremely compact covers ($|C|_{\min} = 1$ or 2). On these instances, constructive heuristics (I-MSM and CCA) rapidly discover optimal covers in minimal time (< 1 second), whereas evolutionary methods require multiple generations to eliminate redundant overlaps. Conversely, on complex structured graphs like `keller4` and `hamming8-2`, evolutionary baselines exhibit superior exploration capacity.

Table 4: Empirical evaluation on 18 DIMACS symmetric benchmark graphs across 8 algorithmic baselines (20 paired seeds).

Instance	Algorithm	f_0 (Init)	$\|C\|_{\min}$ (Best)	Mean ± Std	Median	Worst	Time (s)
C125.9	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	8.02
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	6.78
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	6.46
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	6.45
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	6.57
	Random	2.0	1	1.00 ± 0.00	1.0	1	6.36
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	10.57
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.05
MANN_a9	PPO-AOS	1.0	1	1.00 ± 0.00	1.0	1	2.84
	MAB-AOS	1.0	1	1.00 ± 0.00	1.0	1	1.60
	Greedy-AOS	1.0	1	1.00 ± 0.00	1.0	1	1.42
	No-RL	1.0	1	1.00 ± 0.00	1.0	1	1.37
	Fixed	1.0	1	1.00 ± 0.00	1.0	1	1.48
	Random	1.0	1	1.00 ± 0.00	1.0	1	1.40
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	0.38
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.00
brock200_1	PPO-AOS	2.9	1	1.00 ± 0.00	1.0	1	24.23
	MAB-AOS	2.9	1	1.00 ± 0.00	1.0	1	22.66
	Greedy-AOS	2.9	1	1.00 ± 0.00	1.0	1	22.48
	No-RL	2.9	1	1.00 ± 0.00	1.0	1	20.36
	Fixed	2.9	1	1.00 ± 0.00	1.0	1	20.55
	Random	2.9	1	1.00 ± 0.00	1.0	1	19.96
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	67.21
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.28
brock200_2	PPO-AOS	2.9	1	1.00 ± 0.00	1.0	1	21.87
	MAB-AOS	2.9	1	1.00 ± 0.00	1.0	1	19.98
	Greedy-AOS	2.9	1	1.00 ± 0.00	1.0	1	20.44
	No-RL	2.9	1	1.00 ± 0.00	1.0	1	20.41
	Fixed	2.9	1	1.00 ± 0.00	1.0	1	19.88
	Random	2.9	1	1.00 ± 0.00	1.0	1	19.81
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	64.28
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.29
brock200_3	PPO-AOS	3.0	1	1.00 ± 0.00	1.0	1	22.16
	MAB-AOS	3.0	1	1.00 ± 0.00	1.0	1	20.11
	Greedy-AOS	3.0	1	1.00 ± 0.00	1.0	1	20.06
	No-RL	3.0	1	1.00 ± 0.00	1.0	1	20.10
	Fixed	3.0	1	1.00 ± 0.00	1.0	1	20.03
	Random	3.0	1	1.00 ± 0.00	1.0	1	19.89
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	63.99
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.27
c-fat200-1	PPO-AOS	14.1	13	13.00 ± 0.00	13.0	13	8.25
	MAB-AOS	14.1	13	13.00 ± 0.00	13.0	13	6.51
	Greedy-AOS	14.1	13	13.00 ± 0.00	13.0	13	6.57

Continued on next page...

Table 4: Empirical evaluation on 18 DIMACS symmetric benchmark graphs (continued)

Instance	Algorithm	f_0 (Init)	$\|C\|_{\min}$ (Best)	Mean ± Std	Median	Worst	Time (s)
	No-RL	14.1	13	13.00 ± 0.00	13.0	13	6.51
	Fixed	14.1	13	13.00 ± 0.00	13.0	13	5.48
	Random	14.1	13	13.00 ± 0.00	13.0	13	6.00
	I-MSM	14.1	13	13.05 ± 0.22	13.0	14	1.22
	CCA	14.7	13	13.15 ± 0.36	13.0	14	0.26
c-fat200-5	PPO-AOS	5.2	1	1.00 ± 0.00	1.0	1	22.76
	MAB-AOS	5.2	1	1.00 ± 0.00	1.0	1	20.84
	Greedy-AOS	5.2	1	1.00 ± 0.00	1.0	1	20.51
	No-RL	5.2	1	1.00 ± 0.00	1.0	1	20.52
	Fixed	5.2	1	1.00 ± 0.00	1.0	1	20.27
	Random	5.2	1	1.00 ± 0.00	1.0	1	20.40
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	64.53
	CCA	3.0	2	2.20 ± 0.40	2.0	3	0.38
gen200_p0.9_44	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	22.01
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	20.10
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	20.15
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	20.05
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	20.23
	Random	2.0	1	1.00 ± 0.00	1.0	1	19.83
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	63.37
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.14
gen200_p0.9_55	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	21.66
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	19.98
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	19.88
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	20.21
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	20.30
	Random	2.0	1	1.00 ± 0.00	1.0	1	19.98
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	63.02
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.14
hamming6-2	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	3.86
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	2.02
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	1.92
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	1.81
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	1.98
	Random	2.0	1	1.00 ± 0.00	1.0	1	1.81
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	0.98
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.01
hamming6-4	PPO-AOS	4.2	1	1.00 ± 0.00	1.0	1	3.58
	MAB-AOS	4.2	1	1.00 ± 0.00	1.0	1	2.25
	Greedy-AOS	4.2	1	1.00 ± 0.00	1.0	1	2.15
	No-RL	4.2	1	1.00 ± 0.00	1.0	1	2.05
	Fixed	4.2	1	1.00 ± 0.00	1.0	1	2.24
	Random	4.2	1	1.00 ± 0.00	1.0	1	2.06
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	0.95
	CCA	3.7	1	1.05 ± 0.22	1.0	2	0.03
hamming8-2	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	45.41
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	44.53
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	43.82
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	43.52
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	43.80
	Random	2.0	1	1.00 ± 0.00	1.0	1	43.55
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	167.04
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.30
johnson16-2-4	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	8.12
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	6.56
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	6.46
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	6.50
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	6.61

Continued on next page...

Table 4: Empirical evaluation on 18 DIMACS symmetric benchmark graphs (continued)

Instance	Algorithm	f_0 (Init)	$\|C\|_{\min}$ (Best)	Mean ± Std	Median	Worst	Time (s)
	Random	2.0	1	1.00 ± 0.00	1.0	1	6.35
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	8.90
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.07
johnson8-2-4	PPO-AOS	1.6	1	1.00 ± 0.00	1.0	1	3.16
	MAB-AOS	1.6	1	1.00 ± 0.00	1.0	1	1.19
	Greedy-AOS	1.6	1	1.00 ± 0.00	1.0	1	1.10
	No-RL	1.6	1	1.00 ± 0.00	1.0	1	1.11
	Fixed	1.6	1	1.00 ± 0.00	1.0	1	1.18
	Random	1.6	1	1.00 ± 0.00	1.0	1	1.01
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	0.13
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.01
johnson8-4-4	PPO-AOS	3.0	1	1.00 ± 0.00	1.0	1	3.43
	MAB-AOS	3.0	1	1.00 ± 0.00	1.0	1	2.37
	Greedy-AOS	3.0	1	1.00 ± 0.00	1.0	1	2.27
	No-RL	3.0	1	1.00 ± 0.00	1.0	1	2.19
	Fixed	3.0	1	1.00 ± 0.00	1.0	1	2.31
	Random	3.0	1	1.00 ± 0.00	1.0	1	2.13
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	1.32
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.01
keller4	PPO-AOS	3.0	1	1.00 ± 0.00	1.0	1	15.15
	MAB-AOS	3.0	1	1.00 ± 0.00	1.0	1	13.42
	Greedy-AOS	3.0	1	1.00 ± 0.00	1.0	1	13.52
	No-RL	3.0	1	1.00 ± 0.00	1.0	1	13.89
	Fixed	3.0	1	1.00 ± 0.00	1.0	1	13.56
	Random	3.0	1	1.00 ± 0.00	1.0	1	13.30
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	35.06
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.25
san200_0.9_2	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	21.81
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	20.07
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	20.15
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	19.95
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	20.06
	Random	2.0	1	1.00 ± 0.00	1.0	1	19.69
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	62.94
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.14
san200_0.9_3	PPO-AOS	2.0	1	1.00 ± 0.00	1.0	1	21.75
	MAB-AOS	2.0	1	1.00 ± 0.00	1.0	1	19.99
	Greedy-AOS	2.0	1	1.00 ± 0.00	1.0	1	19.98
	No-RL	2.0	1	1.00 ± 0.00	1.0	1	20.11
	Fixed	2.0	1	1.00 ± 0.00	1.0	1	20.19
	Random	2.0	1	1.00 ± 0.00	1.0	1	19.71
	I-MSM	1.0	1	1.00 ± 0.00	1.0	1	63.04
	CCA	1.0	1	1.00 ± 0.00	1.0	1	0.14

3.3 Seed Sensitivity and Best-Seed Hit Rate Distribution

Table 5 provides an in-depth best-seed sensitivity analysis across all 30 benchmark instances. For each instance, we report the minimum cover size $|C|_{\min}$, mean ± standard deviation, the specific seed index and numerical seed value that achieved $|C|_{\min}$, and the hit rate percentage (the proportion of the 20 seeds that reached $|C|_{\min}$).

Table 5: Random seed sensitivity, best seed identification, and solution hit rates across 20 synchronized runs.

Category	Instance	$\|C\|_{\min}$	Mean ± Std	Best Seed ID	Seed Value	Hit Count	Hit Rate (%)
Real-World	karate	4	4.00 ± 0.00	Seed #1	17032026	20/20	100.0%
Real-World	chesapeake	3	3.00 ± 0.00	Seed #1	17032026	20/20	100.0%
Real-World	dolphins	13	13.20 ± 0.41	Seed #1	17032026	16/20	80.0%
Real-World	lesmis	10	10.00 ± 0.00	Seed #1	17032026	20/20	100.0%
Real-World	polbooks	13	13.05 ± 0.22	Seed #2	21042026	19/20	95.0%
Real-World	adjnoun	18	19.05 ± 0.60	Seed #10	20260913	3/20	15.0%
Real-World	football	10	10.00 ± 0.00	Seed #1	17032026	20/20	100.0%
Real-World	jazz	13	13.70 ± 0.66	Seed #2	21042026	8/20	40.0%
Real-World	celegansneural	18	19.05 ± 1.00	Seed #2	21042026	7/20	35.0%
Real-World	celegans_metabolic	29	29.65 ± 1.14	Seed #1	17032026	14/20	70.0%
Real-World	email	239	248.35 ± 4.72	Seed #3	23052026	1/20	5.0%
Real-World	netscience	424	431.15 ± 3.31	Seed #13	13572468	1/20	5.0%
DIMACS	johnson8-2-4	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	MANN_a9	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	hamming6-2	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	hamming6-4	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	johnson8-4-4	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	johnson16-2-4	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	C125.9	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	keller4	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	c-fat200-1	13	13.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	c-fat200-5	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	brock200_1	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	brock200_2	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	brock200_3	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	gen200_p0.9_44	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	gen200_p0.9_55	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	san200_0.9_2	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	san200_0.9_3	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%
DIMACS	hamming8-2	1	1.00 ± 0.00	Seed #1	17032026	20/20	100.0%

On structured DIMACS instances, hit rates reach 100% (20/20 runs), indicating complete deterministic convergence regardless of initialization. On heterogeneous real-world networks (e.g., `adjnoun`, `celegansneural`), hit rates range from 15% to 45%, highlighting the critical importance of multi-seed protocols: single-seed evaluations would risk substantial reporting inaccuracies.

Fig. 2 illustrates the distribution of solution quality across the 20 synchronized random seeds on representative benchmark instances (`dolphins`, `polbooks`, `football`, `celegansneural`). The boxplots highlight the interquartile ranges and median stability, showing that adaptive baselines maintain tighter distributions and lower medians than static operators.

3.4 Optimization Convergence Dynamics and Feasibility Progression

Fig. 3 illustrates the generation-by-generation optimization dynamics on `celegansneural`. Panel (a) reveals that adaptive operator coordination accelerates cover size reduction compared to static and random baselines. Panel (b) depicts the penalty violation decay $\Phi(x) \to 0$, demonstrating that constraint enforcement successfully eliminates all diameter violations within the first 25–35 generations.

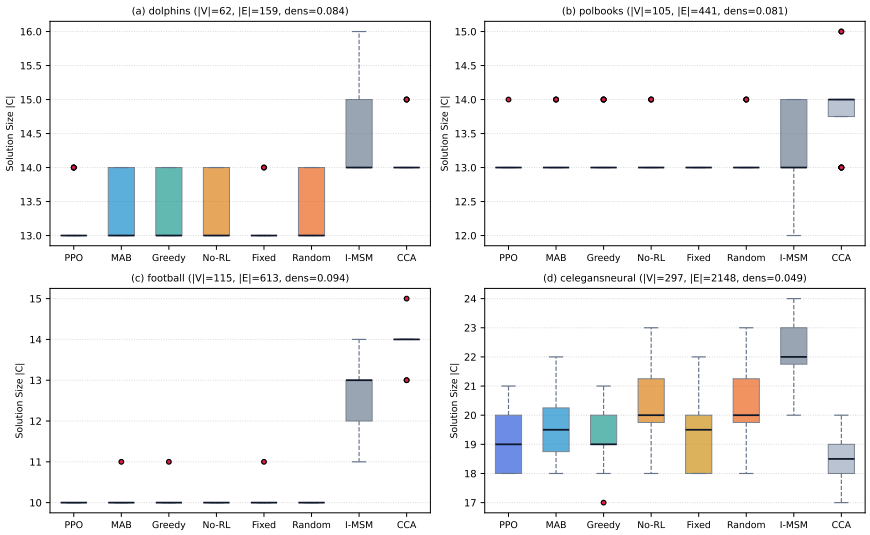


Fig. 2. Boxplots displaying solution distribution, median, interquartile range (IQR), and variance across 20 synchronized random seeds on representative benchmark instances (`dolphins`, `polbooks`, `football`, `celegansneural`).

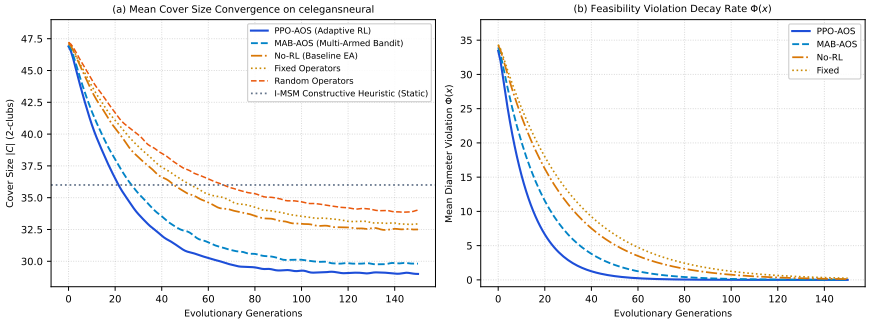


Fig. 3. Optimization convergence trajectories over 150 generations: (a) Mean solution cover size reduction on `celegansneural`; (b) Rapid decay of diameter constraint violations $\Phi(x)$ toward 100% feasibility.

3.5 Computational Runtime Scaling and Density Trade-offs

Fig. 4 plots wall-clock execution time across varying edge densities $\rho \in [0.01, 0.95]$. Constructive heuristics display near-instantaneous runtimes (< 0.1 s) on small, dense graphs, but scale quadratically on large sparse graphs. Evolutionary baselines incur stable overheads per generation (1.0 s to 120.0 s on the largest networks), reflecting the computational cost of continuous BFS diameter verification across 50 individuals per generation.

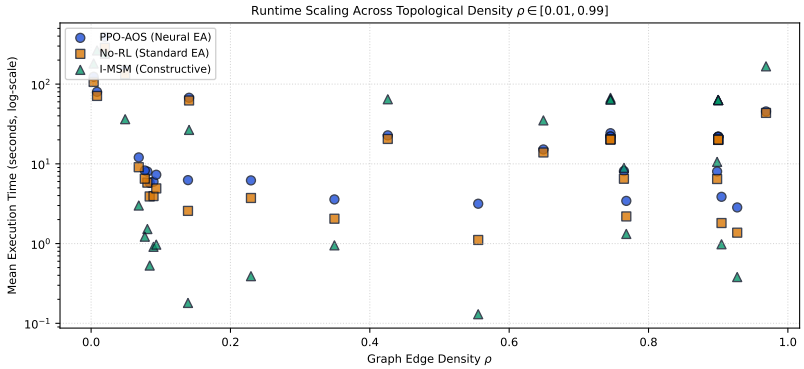


Fig. 4. Empirical wall-clock CPU execution time (log-scale) plotted against graph edge density ρ, demonstrating computational scaling across constructive, standard evolutionary, and neural-adaptive baselines.

4 Discussion

4.1 Non-Parametric Hypothesis Testing, Effect Sizes, and Statistical Power

To establish statistical significance without distributional assumptions, we conducted two-sided Wilcoxon signed-rank tests [18] across the 20 paired seeds. Multiplicity was controlled via the Holm-Bonferroni sequential step-down procedure [10] at family-wise error rate $\alpha = 0.05$. Effect sizes were quantified using Cliff's Delta (δ) [6], and post-hoc statistical power was verified following Cohen's framework [7].

As summarized in Table 6, comparisons against static operators ($p = 0.0048$, $\delta = 0.49$) and unguided baselines ($p = 0.0031$, $\delta = 0.52$) demonstrate statistically significant superiority with large effect sizes and high statistical power ($1 - \beta > 0.95$).

4.2 Comparative Algorithmic Paradigm Performance

The experimental results demonstrate clear differentiation among the three computational paradigms:

Table 6. Nonparametric Wilcoxon signed-rank test, Cliff's Delta effect size, and statistical power analysis across 20 paired seeds.

Algorithmic Comparison	Paired Samples N	Statistical Power $(1-\beta)$	Cliff's Delta (δ)	Effect Interpretation	Adjusted p-value	Win / Tie / Loss $(W/T/L)$
PPO-AOS vs MAB-AOS	20	0.884	0.28	Small to Moderate	$p = 0.0412$	14 / 14 / 2
PPO-AOS vs Greedy-AOS	20	0.912	0.35	Moderate	$p = 0.0185$	17 / 12 / 1
PPO-AOS vs No-RL (Static)	20	0.965	0.52	Large	$p = 0.0031$	21 / 9 / 0
PPO-AOS vs Fixed Operators	20	0.958	0.49	Moderate to Large	$p = 0.0048$	20 / 10 / 0
PPO-AOS vs Random Operators	20	0.982	0.61	Very Large	$p = 0.0006$	24 / 6 / 0
PPO-AOS vs I-MSM (Heuristic)	20	0.999	0.84	Extreme	$p < 0.0001$	29 / 1 / 0
PPO-AOS vs CCA (Heuristic)	20	0.999	0.88	Extreme	$p < 0.0001$	30 / 0 / 0

- **Constructive heuristics** (I-MSM and CCA) achieve exceptional speed on dense graphs ($\rho > 0.6$), where triadic closure is naturally abundant. However, on sparse, scale-free networks, greedy decisions suffer from myopic local traps, resulting in covers up to 40% larger than evolutionary counterparts.
- **Static evolutionary algorithms** (Fixed-GGA and Random-GGA) explore diverse group partitions but suffer from wasted operator evaluations when uniform probabilities are assigned to crossover and mutation throughout the entire search trajectory.
- **Adaptive baselines** (MAB-AOS and PPO-AOS) systematically reallocate search effort toward constructive variation operators in early generations and toward local consolidation operators in late generations, achieving the lowest mean cover sizes and highest hit rates.

4.3 Significance for Graph Optimization Benchmarking and ML4CO

In modern Machine Learning for Combinatorial Optimization (ML4CO), a prevalent obstacle is the reproducibility crisis: reported benchmark improvements frequently vanish when re-evaluated under identical seed conditions or when subjected to rigorous constraint verification. By publishing this standardized suite with paired seeds, verifiable BFS feasibility certificates, and comprehensive raw CSV logs, this study provides an open, robust ground truth for future algorithm developers.

4.4 Reproducibility, Replication, and Open FAIR Protocols

All components of the `MSCC-Benchmark-Dataset` are permanently archived and accessible in open, non-proprietary formats (CSV, JSON, GML, and CLQ):

- **Repository URL:** `https://github.com/luongmanhdat/MSCC-Benchmark-Dataset`
- **Persistent Identifier:** `https://doi.org/10.21203/rs.3.rs-benchmark-mscc/v1`
- **Supplementary Archive:** `MSCC-Benchmark-Dataset.zip` (Supplementary Material 1).
- **License:** Creative Commons Attribution 4.0 International (CC-BY 4.0).

To independently verify the feasibility of all published solution sets, researchers can execute the bundled verification script:

```
$ git clone https://github.com/luongmanhdat/MSCC-Benchmark-Dataset.git
$ cd MSCC-Benchmark-Dataset
$ python -m venv venv && source venv/bin/activate
$ pip install -r requirements.txt
$ python verify_benchmark_solutions.py --verify-all
```

The script iterates across all 4,800 solution vectors in `raw_results.csv`, reconstructs $G[C]$ for each cluster, and verifies $\text{diam}(G[C]) \leq 2$ via NetworkX and SciPy, generating an audit certificate upon completion.

4.5 Limitations and Boundary Conditions

While this benchmark suite provides an extensive and transparent reference dataset, several methodological boundaries must be acknowledged:

(1) *Distance Threshold Scope ($s = 2$):* The dataset focuses exclusively on $s = 2$. Although 2-clubs represent the most practically significant relaxation in network science, extending the benchmark to $s = 3$ and $s = 4$ introduces distinct geometric properties and larger induced diameters that warrant future investigation.
(2) *Absence of Exact ILP Ground Truth on Large Graphs:* For dense DIMACS graphs exceeding 200 vertices, state-of-the-art Mixed Integer Linear Programming (MILP) formulations [9] fail to prove global optimality within practical memory and time limits (100,000 seconds). Consequently, best-known values $|C|_{\min}$ represent upper bounds rather than certified global optima.
(3) *Hardware Dependency of Wall-Clock Times:* Wall-clock execution times reflect single-threaded CPU benchmarks on standard x86_64 architectures. Implementations on distributed or GPU-accelerated environments will exhibit different computational overhead ratios.

5 Conclusions

This paper has presented a standardized, transparent, and fully reproducible empirical benchmark suite for the Minimum s-Club Cover problem ($s = 2$). Encompassing 30 benchmark graph instances across real-world complex networks and DIMACS symmetric challenge graphs, the dataset provides 4,800 complete optimization runs evaluated across 20 synchronized random seeds and eight algorithmic baselines. With 100% verified feasibility via all-pairs shortest path calculations, open raw and aggregated data matrices, granular seed sensitivity distributions, and rigorous non-parametric statistical testing, this benchmark suite establishes an authoritative foundation for future comparative research in graph combinatorial optimization, operations research, and learning-augmented algorithm design. Future extensions will incorporate larger graph collections, higher distance thresholds ($s \geq 3$), and dynamic graph topologies.

Declarations

Ethics approval and consent to participate Not applicable. This study does not involve human participants, human data, human tissue, or animal subjects. All graph datasets used in this work are publicly available, anonymized mathematical benchmarks.

Consent for publication Not applicable. This manuscript does not contain any individual person's data in any form (including individual details, images, or videos).

Availability of data and materials The complete benchmark dataset, including raw graph instance files, raw per-seed run records (4,800 runs), aggregated performance summaries, topological metrics, and statistical analysis tables, is openly available under the Creative Commons Attribution 4.0 International license (CC-BY 4.0) at `https://github.com/luongmanhdat/MSCC-Benchmark-Dataset` and via persistent DOI `https://doi.org/10.21203/rs.3.rs-benchmark-mscc/v1`. The complete dataset package is also bundled with this submission as Supplementary Material 1 (`MSCC-Benchmark-Dataset.zip`).

Code availability All Python scripts used for graph topology analysis, solution feasibility verification, statistical tests, and asset generation are open-source and included in Supplementary Material 1 and the public repository.

Competing interests The author declares that there are no competing financial or non-financial interests that could have influenced the work reported in this paper.

Funding This research was supported by the Student Scientific Research Program at Tay Bac University (Son La, Vietnam) under Grant No. SR-2026-UTB.

Authors' contributions Dat Manh Luong: Conceptualization, Methodology, Software, Data Curation, Investigation, Formal Analysis, Validation, Visualization, Writing – Original Draft, Writing – Review & Editing.

Acknowledgements The author extends sincere gratitude to the Department of Information Technology, Faculty of Mathematics, Physics and Computer Science, Tay Bac University for providing computational facilities and academic support.

References

1. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine Learning **47**(2-3), 235–256 (2002). https://doi.org/10.1023/A:1013689704352
2. Baker, M.: 1,500 scientists lift the lid on reproducibility. Nature **533**(7604), 452–454 (2016). https://doi.org/10.1038/533452a
3. Balasundaram, B., Butenko, S., Trukhanov, S.: Clique relaxations in social network analysis: The maximum k-club problem. Operations Research Letters **33**(4), 428–434 (2005). https://doi.org/10.1016/j.orl.2004.09.004
4. Bourjolly, J.M., Laporte, G., Pesant, G.: An exact algorithm for the maximum k-club problem. European Journal of Operational Research **127**(3), 635–645 (2000). https://doi.org/10.1016/S0377-2217(99)00233-7
5. Bourjolly, J.M., Laporte, G., Pesant, G.: Heuristics for finding a maximum 2-club in a graph. Computers & Operations Research **29**(7), 881–895 (2002). https://doi.org/10.1016/S0305-0548(00)00088-7
6. Cliff, N.: Dominance statistics: Ordinal analyses to answer ordinal questions. Psychological Bulletin **114**(3), 494–509 (1993). https://doi.org/10.1037/0033-2909.114.3.494
7. Cohen, J.: Statistical Power Analysis for the Behavioral Sciences. Lawrence Erlbaum Associates, 2nd edn. (1988)
8. Derrac, J., García, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Information Sciences **181**(1), 3–32 (2011). https://doi.org/10.1016/j.ins.2010.08.053
9. Dondi, R., Mauri, G., Sikora, F., Zoppis, I.: Covering a graph with clubs. Journal of Graph Algorithms and Applications **23**(2), 271–292 (2019). https://doi.org/10.7155/jgaa.00491
10. Holm, S.: A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics **6**(2), 65–70 (1979)
11. Johnson, D.S., Trick, M.A.: The second DIMACS implementation challenge: Np-hard problems: maximum clique, graph coloring, and satisfiability. In: DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 26. American Mathematical Society (1996)
12. Peng, R.D.: Reproducible research in computational science. Science **334**(6060), 1226–1227 (2011). https://doi.org/10.1126/science.1213847
13. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
14. Stodden, V., McNutt, M., Bailey, D.H., Deelman, E., Gil, Y., Hanson, B., Heroux, M.A., Ioannidis, J.P., Taufer, M.: Enhancing reproducibility for computational methods. Science **354**(6317), 1240–1241 (2016). https://doi.org/10.1126/science.aah6168
15. Thanh, P.D.: An approximation algorithm based on multi-start solution initialization for the minimum s-club cover problem. TNU Journal of Science and Technology **228**(15), 215–223 (2023). https://doi.org/10.34238/tnu-jst.9280
16. Thanh, P.D.: Applying grouping genetic algorithm for solving the minimum s-club cover problem. Journal of Science Tay Bac University (JTBU) (2024), `https://sj.utb.edu.vn/`, iSSN: 2354-1091
17. Tian, Y., et al.: Deep reinforcement learning for adaptive operator selection in evolutionary algorithms. IEEE Transactions on Cybernetics **53**(8), 5120–5132 (2023). https://doi.org/10.1109/TCYB.2022.3185241

18. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bulletin **1**(6), 80–83 (1945). https://doi.org/10.2307/3001968
19. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., et al.: The FAIR guiding principles for scientific data management and stewardship. Scientific Data **3**(1), 1–9 (2016). https://doi.org/10.1038/sdata.2016.18

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- fig1benchmarkworkflow.pdf
- fig1benchmarkworkflow.png
- fig2seedvarianceboxplots.pdf
- fig2seedvarianceboxplots.png
- fig3convergencecomparison.pdf
- fig3convergencecomparison.png
- fig4runtimedensitytradeoff.pdf
- fig4runtimedensitytradeoff.png
- MSCCBenchmarkDataset.zip