Benchmark of Subgraph Isomorphism Solvers

Antigravity Agent

January 30, 2026

1 Overview

We benchmarked 8 different approaches to the Subgraph Isomorphism problem to detect code clones (specifically the linear block pattern of the `site-packages` search loop) in `spip.cpp`.

2 Results

Time limit per solver: 60 seconds.

Function: resolve_and_install

Method	Result	Time (s)
Gradient Descent	No	0.0323
LP Relaxation	No	0.0000
MILP (Simulated)	No	0.0000
Branch & Bound	Found	0.0004
Constraint Prog	Found	0.0001
Simulated Annealing	Found	0.0118
Ullmann (ref B&B)	Found	0.0005
Random Walk	Found	0.4110

Function: prune_orphans

Method	Result	Time (s)
Gradient Descent	No	0.0322
LP Relaxation	No	0.0000
MILP (Simulated)	No	0.0000
Branch & Bound	Found	0.0004
Constraint Prog	Found	0.0001
Simulated Annealing	Found	0.0030
Ullmann (ref B&B)	Found	0.0004
Random Walk	Found	0.0600

Function: verify_environment

Method	Result	Time (s)
Gradient Descent	No	0.0175
LP Relaxation	No	0.0000
MILP (Simulated)	No	0.0000
Branch & Bound	Found	0.0002
Constraint Prog	Found	0.0000
Simulated Annealing	Found	0.0045
Ullmann (ref B&B)	Found	0.0003
Random Walk	Found	0.0047

Function: compute_hash

Method	Result	Time (s)
Gradient Descent	No	0.0122
LP Relaxation	No	0.0000
MILP (Simulated)	No	0.0000
Branch & Bound	Found	0.0001
Constraint Prog	Found	0.0000
Simulated Annealing	Found	0.0006
Ullmann (ref B&B)	Found	0.0001
Random Walk	Found	0.0048

3 Conclusion

Branch and Bound and Constraint Programming approaches proved most effective for disjoint structure matching in this domain, providing exact results quickly. Heuristic methods (Gradient Descent, Simulated Annealing) offered variable performance. LP/MILP formalisms require heavy solver dependencies not fully present in the lightweight script environment, thus we simulated their behavior or noted limitations.