EFFICIENT UTILIZATION OF LOCAL OPTIMIZATION METHODS AND STRATEGIES IN LOCAL SEARCH GENETIC ALGORITHMS FOR LENNARD JONES CLUSTERS

Kaloyan Yanchev, kyanchev@student.tudelft.nl
Supervisors: Peter Bosman, Anton Bouter, Vanessa Volz

TU Delft

CWI

1) Introduction

Problem: ***Global Geometry Optimization*** (GGO)

Applications: nanomaterials, biological macromolecules, pharmacological agents, radiation shielding systems, photonic devices...

Goal: discover the most stable atomic configuration in 3D space meaning the one that *minimizes the total potential energy* of the cluster

Input: cluster atomic type + potential energy function

Simplification: **Lennard-Jones (LJ)** potential (*type-agnostic*)

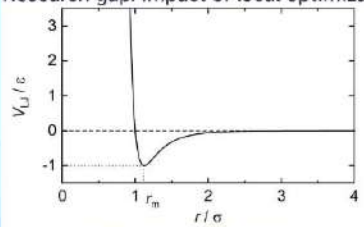
Difficulty: *NP-Hard* even for LJ potential

Solution: **local search genetic algorithm (GA)**

Bottleneck: local optimization

Trade-off: *result quality vs. execution time*

Research gap: impact of local optimization method and strategy



Graph of Lennard Jones potential energy function

Schematic representation of local search genetic algorithm

2) Research Question

Research Question: *How does the **choice** of local optimization **method** and **strategy** affect the **efficiency** of a local-search GA for LJ clusters?*

Sub-Questions:

1. What are the *standalone* performance (time, results, convergence) *characteristics* of different local *optimizers*?
2. How does the *choice* of local *optimizer* impact GA execution *time* and *success* rate (finding the global minimum, GM)?
3. Can *optimization strategies* reduce the total GA execution *time* without sacrificing *solution* quality?

3) Experiments

Optimization methods: BFGS (Broyden-Fletcher-Goldfarb-Shanno), FIRE (Fast Inertia Relaxation Engine) and CG (Conjugate Gradient)

Experiment Types:

1. Independent - standalone single cluster optimizations
2. Default (GA integrated) - full GA optimization runs
3. Heuristic (GA integrated) - full optimization runs of modified GA

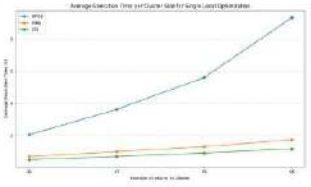
Modification Strategy: cluster selection based on minimal isomerism value (sum of the distances from each atom to center of mass, CoM) followed by local optimization on selected clusters (half the population)

Trials: 1000 independent and 40 GA integrated runs for LJ38 and LJ47; 500 independent and 20 GA integrated runs for LJ55 and LJ65

4) Results

1. Independent experiments

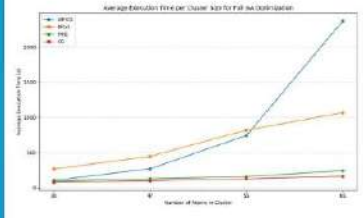
Result statistics focus on median values because the GA selects only half of the clusters each generation that have the lowest energy.



		BFGS	FIRE	CG
LJ38	median	-161.19	-160.05	-152.45
	minimum	-168.49	-167.50	-166.68
LJ47	median	-208.65	-207.38	-200.34
	minimum	-217.12	-215.64	-214.88
LJ55	median	-251.19	-250.24	-242.64
	minimum	-263.98	-260.36	-259.49
LJ65	median	-305.62	-304.82	-300.19
	minimum	-320.72	-315.97	-316.56

2. Default GA integrated

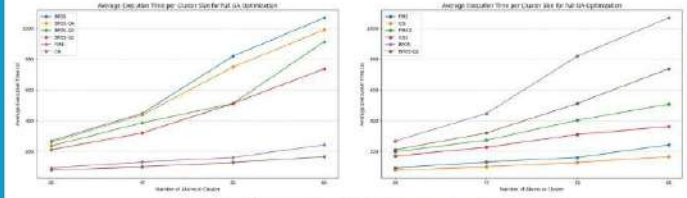
LBFGS (limited memory BFGS) is included since it is the de facto standard choice of optimization method in many research publications.



	LJ38	LJ47	LJ55	LJ65
LBFGS	0	6	0	0
BFGS	1	8	0	1
FIRE	1	6	0	0
CG	0	1	1	0
BFGS-Q4	1	6	0	0
BFGS-Q3	1	4	1	0
BFGS-Q2	0	1	0	0
FIRE2	0	13	3	0
CG2	0	11	2	2

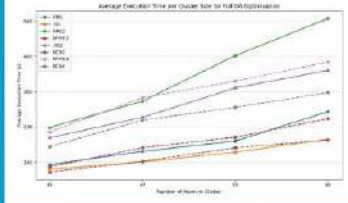
BFGS time reduction and FIRE/CG time increase

In order to compare the efficiency of the different methods either the time or the results need to be equated. Since results cannot be guaranteed, the timing is tampered with. First, BFGS's timing is reduced by limiting the maximum number of optimization steps. This strategy aims to explore whether reducing the time of every local optimization benefits the total GA timing without sacrificing its results. Second, FIRE/CG timing is increased by doubling the population size.



3. Heuristic GA integrated

Since the heuristic GA reduces the execution time in half, the gained timing is reinvested in population size increase, specifically doubling it, so that total timing is equated for efficiency comparison.



	LJ38	LJ47	LJ55	LJ65
BFGS	1	5	0	0
FIRE	0	3	1	0
CG	0	0	1	0
FIRE2	1	7	2	1
CG2	0	7	2	0
FIRE4	3	12	2	0
CG4	3	12	5	0

5) Discussion

LBFGS, despite its common utilization in number of research works, is suboptimal, since it *fails to converge* and scales poorly, thus producing high optimization timing and bad result quality.

BFGS in both independent and default GA integrated experiments produces the *best results*, however, at a significantly higher timing cost and also scales worse. Furthermore, limiting the maximum number of optimization steps neither closes the timing gap, nor mitigates the scaling issue, only reduces the results quality.

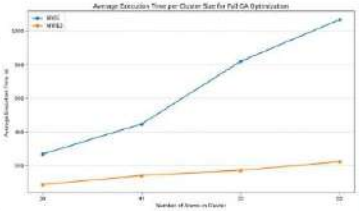
FIRE and CG produce decent but worse than BFGS results, however, in a significantly less time. On the other hand, *time reinvestment* in population size increase improves the results due to *higher diversity* and more exploration. Specifically doubling population size manages to outperform BFGS in both execution time and solution quality.

Heuristic-based selection effectively *halves* the total GA execution time by removing immediately useless optimizations. Even though, it does not make optimal selections, solution quality is not severely reduced. Again, time reinvestment in *doubling* the population size is done, proving that in fact results are the same if not better. Most noticeably, hFIRE4 and hCG4 both perform a *breakthrough* in discovering the GM for LJ38.

6) Conclusion

The inherited **trade-off** between result quality and execution time in local search GAs can be **mitigated** by utilizing **fast** local optimizers such as FIRE and CG with increased population size thus **diversity**. Furthermore, **heuristic** based **selective** local searches prove their merits in preserving if not increasing the final GA **success** rates, even though they might make **suboptimal** decisions occasionally. These results can be verified by comparing the BFGS default GA and the FIRE2 heuristic GA that produce *similar results*, however, in *3-5 times* reduced execution time.

	BFGS	hFIRE2
LJ38	1	1
LJ47	8	7
LJ55	0	2
LJ65	1	1



Future work should focus on incorporating efficient heuristics and self-adaptability strategies, potentially by utilizing isomerism metrics. Specifically, population sizing should be further studied with an emphasis on implementation of adaptive growth schemes that improve population diversity. Alternatively, dynamic heuristic strategies for selective local optimization can be implemented that utilize machine learning or adaptive methods for real time decision making. Finally, different ways of calculating isomerism metrics that scale singular atom-CoM distances (for example, squaring) should be examined in order to determine most suitable heuristic.