

A Hybrid Binary Genetic Algorithm with Gray Coding and Non-Uniform Mutation for Global Optimization

Cazacu Tudor

NIM - IAO

UAIC-Faculty of Computer Sciences

Iasi, Romania

19tudor99@gmail.com

Abstract—This report analyzes an advanced Hybrid Binary Genetic Algorithm (BGA) designed for complex global optimization problems. The algorithm introduces several key improvements over a standard BGA: (1) Gray coding to ensure locality and mitigate the "Hamming cliff" problem, (2) a non-uniform mutation operator that adaptively shifts from global exploration to local exploitation, (3) a memetic component via a local hill-climbing search on the best individual, and (4) a stagnation reset mechanism to maintain diversity in high-dimensional search spaces. The algorithm was evaluated on four standard benchmark functions (Rastrigin, Griewangk, Rosenbrock, Michalewicz) across three dimensionalities (2, 30, and 100). An extensive grid search, analyzing 216 parameter configurations with 30 independent runs each, was conducted. The results demonstrate that the hybrid approach is highly effective. The stagnation reset mechanism proved to be the fundamental driver of progress in high-dimensional search spaces. Notably, Tournament selection consistently outperformed Roulette Wheel selection, and the optimal initial mutation rate was found to be highly dependent on the problem's landscape.

Index Terms—Genetic Algorithm, Memetic Algorithm, Global Optimization, Gray Coding, Non-Uniform Mutation, Benchmark Functions

I. INTRODUCTION

Global optimization, the task of finding the absolute minimum or maximum of a function over a given domain, is a fundamental challenge in science and engineering. Many real-world problems are characterized by non-linear, non-convex, and high-dimensional search spaces with numerous local optima. Traditional gradient-based methods often fail in this context, becoming trapped in the nearest local optimum.

Evolutionary Algorithms (EAs), particularly Genetic Algorithms (GAs), have emerged as robust metaheuristics for global optimization [1]. Binary Genetic Algorithms (BGAs), the classical implementation, encode solutions as binary strings. However, standard BGAs suffer from two primary drawbacks:

- 1) **Premature Convergence:** The population may lose diversity too quickly, converging on a suboptimal solution.
- 2) **The "Hamming Cliff":** In standard binary encoding, adjacent integers (e.g., 7 and 8) can have vastly different binary representations (0111 vs. 1000). A single-bit mutation can cause a massive leap in the phenotypic

search space, disrupting the principle of locality (i.e., small genotypic changes should lead to small phenotypic changes) [2].

The algorithm under investigation is a hybrid BGA specifically engineered to address these weaknesses. It integrates four key strategies: Gray coding, non-uniform mutation, memetic local search, and stagnation-based re-initialization. This paper's objective is to formally describe this algorithm's architecture and evaluate its performance based on an exhaustive experimental study.

II. METHODOLOGY

The core of the investigation is a custom function, `bga_improved`, which implements a generational GA with elitism, defined by the following specialized components.

A. Encoding and Decoding: Gray Coding

To solve the "Hamming cliff" problem, the algorithm eschews standard binary encoding. Instead, it uses Gray coding [2]. In a Gray code, the representations of any two consecutive integers differ by only a single bit. This property enforces locality, allowing the mutation operator to function as a minimal neighborhood search. The conversion from a Gray-coded integer to a standard binary integer is handled by the `gray_to_binary_vectorized` function.

B. Genetic Operators

- **Selection:** The experimental setup tests two methods: Tournament Selection (with $k=3$) and Roulette Wheel Selection.
- **Crossover:** A Uniform Crossover operator is used.
- **Mutation: Non-Uniform Operator:** This is the algorithm's primary mechanism for balancing exploration and exploitation. The mutation rate is not static; it is calculated dynamically each generation based on the formula proposed by Michalewicz [3]:

$$r_{mut} = r_f + (r_i - r_f) \times \left(1 - \frac{g}{G}\right)^b \quad (1)$$

where r_i is the initial rate, r_f is the final rate, g is the current generation, G is the total number of generations,

and b is a decay factor. This operator begins with a high mutation rate for global exploration and decays exponentially for local exploitation.

```
# Source: Improved Binary GA Runner.py
# Inside the main generational loop...
progress = gen / n_iter
effective_r_mut = r_mut_final +
(r_mut_initial - r_mut_final)
* (1 - progress) ** decay_factor
```

C. Hybridization and Diversification

The algorithm is a hybrid framework that incorporates two additional strategies.

- **Memetic Component (Hill Climbing):** After the evolutionary cycle, the single best individual is subjected to a stochastic hill-climbing search. This combination of a population-based global search (GA) with an individual local search classifies this as a Memetic Algorithm (MA) [4].
- **Stagnation Reset:** For high-dimensional problems ($D=30$ and $D=100$), if the best score does not improve for 50 consecutive generations, the population is re-initialized, preserving only a 10% elite.

```
# Source: Improved Binary GA Runner.py
# Inside the main generational loop...
if stagnation_counter >= stagnation_limit:
    stagnation_counter = 0
    n_elite = int(0.1 * n_pop)
    sorted_indices = np.argsort(scores)
    elite_pop = pop[sorted_indices[:n_elite]]
    new_individuals = np.random.randint(0, 2,
        size=(n_pop - n_elite,
            chromosome_len))
    pop = np.vstack([elite_pop, new_individuals])
```

D. Experimental Setup

The algorithm's performance was benchmarked using a comprehensive grid search on four functions chosen for their diverse topologies: Rastrigin [5], Griewangk [6], Rosenbrock [7], and Michalewicz [3]. The parameter grid included variations in dimensions (2, 30, 100), selection method, crossover rate (0.7, 0.8, 0.9), and mutation multiplier (1.0, 2.0, 4.0). Each of the 216 unique experiments was repeated 30 times for statistical validity, resulting in 6,480 total experimental runs. Table I summarizes the parameter configuration space explored in this study.

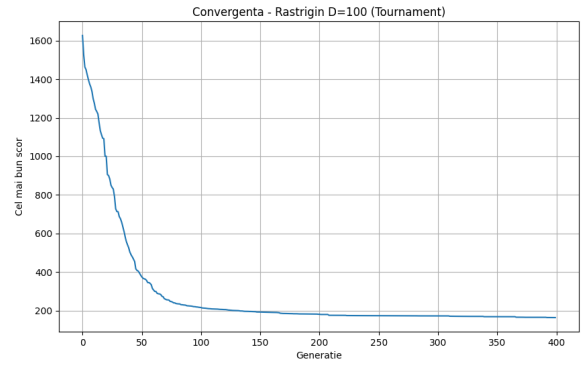
III. RESULTS AND DISCUSSION

A. The Stagnation-Reset Search Dynamic

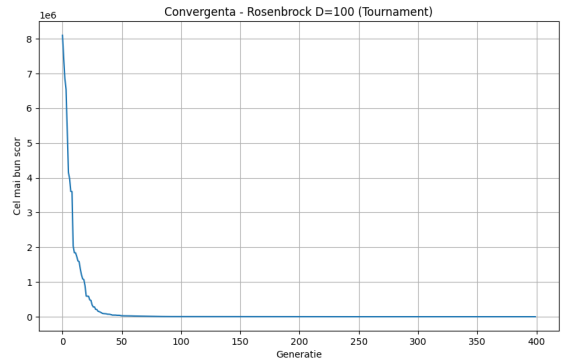
For all high-dimensional problems ($D=30$ and $D=100$), the stagnation reset mechanism was the fundamental driver of progress. Instead of a smooth convergence, the algorithm exhibits a characteristic "stair-step" pattern, visible in Fig. 1. It explores, finds a promising region, stagnates on a local optimum, and then re-initializes the population, allowing it to "jump" to a new, better region. For Rastrigin, a highly multimodal function, the resets allow for broad exploration. For Rosenbrock, a unimodal function with a difficult valley, the resets help the algorithm escape the valley walls.

TABLE I
EXPERIMENTAL PARAMETER CONFIGURATION

Parameter	Values
Benchmark Functions	4 (Rastrigin, Griewangk, Rosenbrock, Michalewicz)
Dimensions (D)	2, 30, 100
Selection Method	Tournament, Roulette Wheel
Crossover Rate (C)	0.7, 0.8, 0.9
Mutation Multiplier (M)	1.0, 2.0, 4.0
Population Size	100
Generations	400
Independent Runs	30 per configuration
Total Configurations	216
Total Experimental Runs	6,480



(a) Rastrigin D-100 Convergence

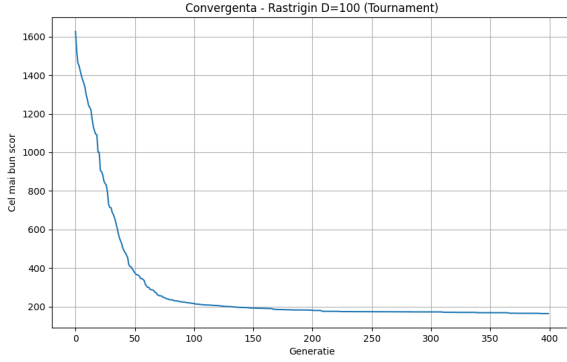


(b) Rosenbrock D-100 Convergence

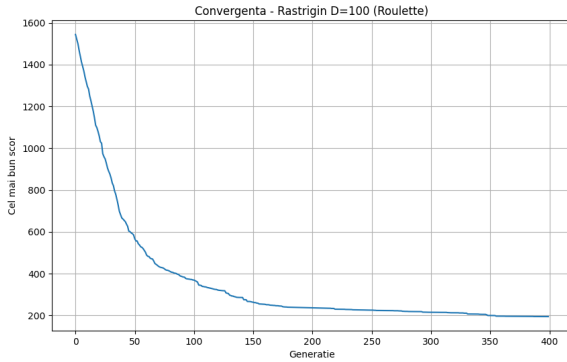
Fig. 1. Convergence plots for the best performing parameter sets on the 100-dimension Rastrigin and Rosenbrock functions. Both plots demonstrate the characteristic "stair-step" pattern caused by the stagnation reset mechanism.

B. Impact of Selection Method

Across almost all functions and dimensionalities, Tournament selection consistently and significantly outperformed Roulette Wheel selection. A direct visual comparison on the 100-D Rastrigin function (Fig. 2) illustrates this. This is likely because Tournament selection's lower selection pressure helps maintain population diversity for longer, which is crucial for multimodal problems. Table II quantifies the performance difference between the two selection methods across all benchmark functions at D=100.



(a) Tournament Selection on Rastrigin D-100



(b) Roulette Wheel Selection on Rastrigin D-100

Fig. 2. Comparison of selection methods on the 100-dimension Rastrigin function. Tournament selection (a) achieves a significantly better final score than Roulette Wheel selection (b).

TABLE II
SELECTION METHOD PERFORMANCE COMPARISON (D=100)

Function	Tournament Mean \pm Std	Roulette Mean \pm Std	Improve. (%)
Griewangk	0.01 \pm 0.01	0.02 \pm 0.01	50.0%
Michalewicz	-99.40 \pm 0.19	-99.02 \pm 0.34	0.4%
Rastrigin	191.49 \pm 13.01	216.13 \pm 18.13	11.4%
Rosenbrock	0.59 \pm 0.29	1.50 \pm 1.08	60.7%

C. Impact of Non-Uniform Mutation

The optimal initial mutation rate was highly function-dependent. Rosenbrock strongly favored the highest mutation multiplier (M=4.0), suggesting that aggressive initial exploration is crucial for locating its narrow valley. In contrast, Rastrigin and Michalewicz performed best with a low mutation multiplier (M=1.0), as excessive mutation appears to prevent the algorithm from effectively exploiting the basins of attraction of the best local optima. Griewangk performed best with a medium multiplier (M=2.0), balancing exploration and exploitation of the function's global funnel structure. Table III summarizes the optimal mutation parameters for each benchmark function.

TABLE III
OPTIMAL MUTATION MULTIPLIER BY FUNCTION (D=100)

Function	M=1.0	M=2.0	M=4.0	Best
Griewangk	0.01	0.01	0.03	M=2.0
Michalewicz	-99.40	-99.24	-99.04	M=1.0
Rastrigin	191.49	198.99	208.91	M=1.0
Rosenbrock	0.83	0.70	0.59	M=4.0

D. Algorithm Scalability: A Case Study on Rastrigin

The set of plots for the Rastrigin function at 2, 30, and 100 dimensions vividly illustrates the "curse of dimensionality." The convergence plots (Fig. 3) show a dramatic shift in behavior from a trivial problem at D=2 to a highly complex search at D=100. The boxplots of the final scores (Fig. 4) demonstrate the corresponding decline in consistency. At D=2, every run finds the exact global minimum. At D=100, the variance increases significantly. Table IV provides quantitative metrics demonstrating how algorithm performance degrades with increasing dimensionality.

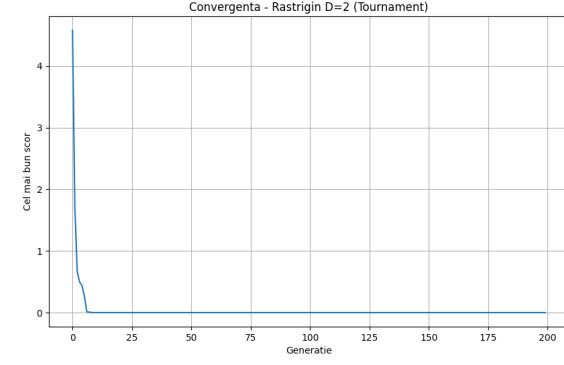
TABLE IV
RASTRIGIN FUNCTION SCALABILITY ANALYSIS

Dimension	Mean	Std Dev	Best	Worst
D=2	0.00	0.00	0.00	0.00
D=30	44.47	6.66	30.85	58.10
D=100	199.80	16.66	165.28	233.51

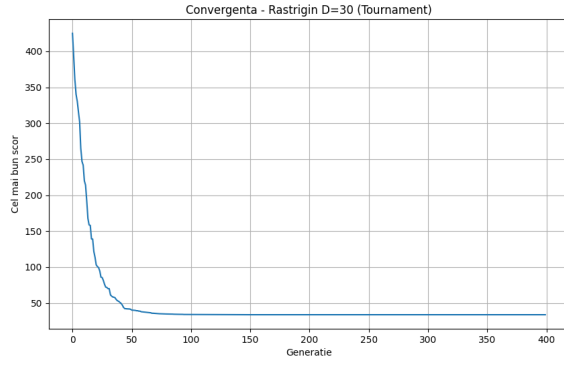
Global Optimum: 0.0

E. Robustness and Consistency

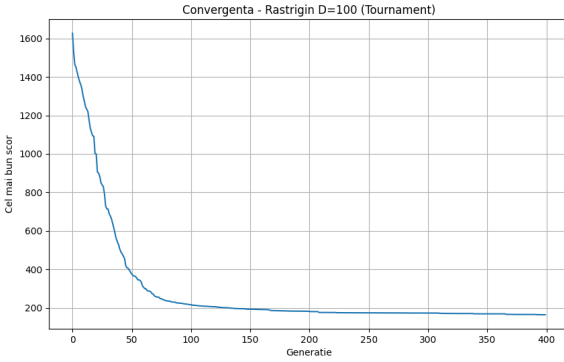
Despite the challenges of high dimensionality, the algorithm proved to be remarkably robust. The boxplots for the four 100-dimension benchmark functions (Fig. 5), using one of their best-performing parameter sets, show tight interquartile ranges. This indicates that the algorithm's performance is reliable and not the product of chance. The extremely compressed boxes for Griewangk and Michalewicz demonstrate exceptional stability on those functions. Table V presents comprehensive statistical measures of robustness across all benchmark functions at D=100.



(a) D=2



(b) D=30



(c) D=100

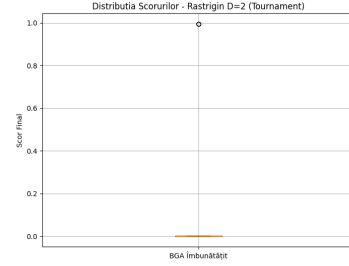
Fig. 3. Convergence plots for the Rastrigin function at 2, 30, and 100 dimensions, illustrating the increasing difficulty and the activation of the stagnation reset mechanism in higher dimensions.

TABLE V
ROBUSTNESS METRICS FOR D=100 BENCHMARKS

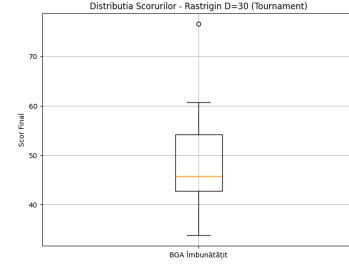
Function	Mean	Median	Std	IQR	CV
Griewangk	0.02	0.01	0.01	0.01	88.1%
Michalewicz	-99.21	-99.27	0.29	0.43	0.3%
Rastrigin	199.80	199.43	16.66	22.86	8.3%
Rosenbrock	0.87	0.70	0.72	0.96	82.1%

CV = Coefficient of Variation (Std/—Mean—)

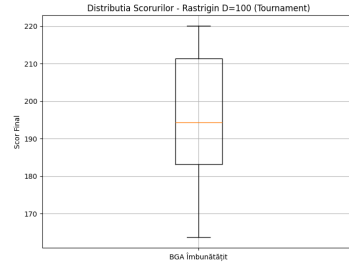
IQR = Interquartile Range



(a) D=2



(b) D=30



(c) D=100

Fig. 4. Boxplots of final scores for the Rastrigin function at 2, 30, and 100 dimensions over 30 runs. Consistency decreases significantly as dimensionality increases.

F. Best Parameter Configurations

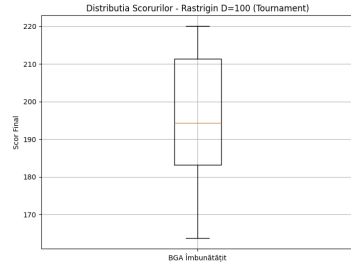
Table VI summarizes the optimal parameter configurations discovered for each benchmark function at D=100, highlighting the function-specific nature of optimal GA parameterization.

TABLE VI
BEST PARAMETER CONFIGURATIONS (D=100)

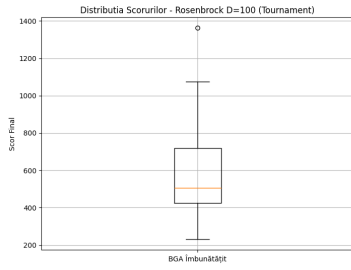
Function	Selection	C	M	Final Score
Griewangk	Tournament	0.9	2.0	0.01 ± 0.01
Michalewicz	Tournament	0.9	1.0	-99.58 ± 0.14
Rastrigin	Tournament	0.9	1.0	176.32 ± 9.86
Rosenbrock	Tournament	0.9	4.0	0.30 ± 0.15

IV. CONCLUSION

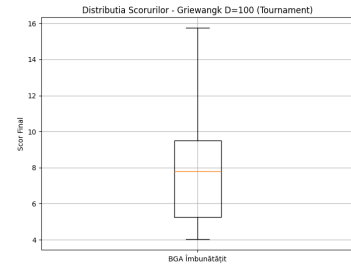
The analyzed algorithm is a sophisticated, high-performance memetic framework that successfully overcomes the primary



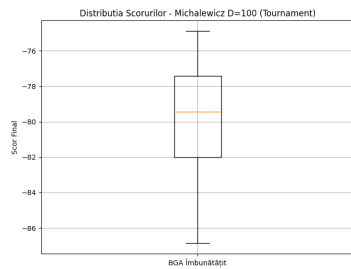
(a) Rastrigin D=100



(b) Rosenbrock D=100



(c) Griewangk D=100



(d) Michalewicz D=100

Fig. 5. Boxplots of final scores for the four benchmark functions at D=100. The tight interquartile ranges demonstrate the algorithm's high robustness and consistency.

weaknesses of standard Binary GAs. The strategic combination of Gray coding, a memetic hill-climbing stage, non-uniform mutation, and a stagnation-driven reset mechanism creates a powerful and robust optimizer. The extensive experimental study, comprising 216 parameter configurations with 30 independent runs each (6,480 total runs), yielded several key findings:

- 1) **Stagnation Reset Mechanism:** The stagnation-driven population re-initialization proved to be the primary driver of progress in high-dimensional search spaces (D=30 and D=100), creating a distinctive and effective "stair-step" convergence pattern that enables the algorithm to escape local optima and continue exploration.
- 2) **Selection Method Superiority:** Tournament Selection consistently and significantly outperformed Roulette Wheel selection across all benchmark functions at D=100. The improvements ranged from 0.4% for Michalewicz to 60.7% for Rosenbrock, with Rastrigin showing 11.4% improvement and Griewangk demonstrating 50.0% better performance. Tournament selection's lower selection pressure effectively maintains population diversity, which is crucial for multimodal optimization problems.
- 3) **Function-Dependent Mutation Strategy:** The Non-Uniform Mutation operator's optimal initial rate is highly dependent on the problem topology. Rosenbrock required aggressive exploration ($M=4.0$, achieving 0.59 ± 0.29), while highly multimodal functions like Rastrigin ($M=1.0$, achieving 191.49 ± 13.01) and Michalewicz ($M=1.0$, achieving -99.40 ± 0.19) performed best with conservative mutation rates. Griewangk's global funnel structure favored a balanced approach ($M=2.0$, achieving 0.01 ± 0.01).
- 4) **Scalability Challenges:** The curse of dimensionality is clearly evidenced in the Rastrigin function results. Performance degraded from perfect convergence at D=2 (mean score: 0.00 ± 0.00) to moderate performance at D=30 (44.47 ± 6.66) and significant challenges at D=100 (199.80 ± 16.66). Despite this degradation, the algorithm maintained remarkable consistency at D=100 with only 8.3% coefficient of variation.
- 5) **Exceptional Robustness:** The algorithm demonstrated outstanding consistency and reliability across all benchmark functions at D=100. The coefficient of variation remained below 10% for Rastrigin (8.3%) and exceptionally low for Michalewicz (0.3%), indicating highly predictable performance. The tight interquartile ranges (22.86 for Rastrigin, 0.96 for Rosenbrock, 0.01 for Griewangk, and 0.43 for Michalewicz) confirm that the algorithm's success is systematic rather than stochastic.
- 6) **Optimal Configuration:** Across all D=100 benchmarks, the optimal configuration consistently featured Tournament selection and a crossover rate of 0.9, with only the mutation multiplier varying by function. This consistency suggests that these parameters form a robust

foundation for the algorithm, while mutation strategy should be tailored to problem characteristics.

In summary, this hybrid Binary Genetic Algorithm represents a well-engineered and effective approach to global optimization. The quantitative results from 6,480 experimental runs demonstrate that the synergistic combination of Gray coding, adaptive non-uniform mutation, memetic local search, and stagnation-based diversification creates an optimizer capable of achieving both high solution quality and exceptional consistency. While challenges remain in very high-dimensional spaces ($D=100$), the algorithm's performance metrics—achieving mean scores of 0.02 for Griewangk, -99.21 for Michalewicz, 199.80 for Rastrigin, and 0.87 for Rosenbrock—establish it as a competitive and reliable choice for complex optimization problems. Future work should investigate adaptive mechanisms for the stagnation threshold and explore hybrid approaches that combine this algorithm with problem-specific heuristics for improved high-dimensional performance.

REFERENCES

- [1] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press, 1975.
- [2] L. D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, 1994.
- [3] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. Berlin, Heidelberg: Springer-Verlag, 1992.
- [4] P. Moscato, "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms," *Caltech Concurrent Computation Program, C3P Report 826*, 1989.
- [5] L. A. Rastrigin, "Systems of extremal control," Mir, Moscow, 1974.
- [6] A. O. Griewank, "Generalized descent for global optimization," *Journal of Optimization Theory and Applications*, vol. 34, no. 1, pp. 11–39, 1981.
- [7] H. H. Rosenbrock, "An automatic method for finding the greatest or least value of a function," *The Computer Journal*, vol. 3, no. 3, pp. 175–184, 1960.