# Comparative Analysis of Genetic Algorithm Configurations for Continuous Optimization Functions

Apăvăloaiei Alexandru
Babeș-Bolyai University
Computer Science Department

June 2, 2025

# Contents

# 1 Introduction

This study presents a comprehensive comparison of different genetic algorithm (GA) configurations applied to two continuous optimization benchmark functions. The research investigates the performance of various representation schemes and crossover operators under controlled experimental conditions with fixed computational budgets.

The primary objectives of this study are:

- Evaluate the effectiveness of binary vs. real-valued representations
- Compare different crossover operators within each representation scheme
- Analyze statistical significance of performance differences
- Provide recommendations for GA configuration selection

# 2 Test Functions

## 2.1 Drop Wave Function

The Drop Wave function is a multimodal optimization function commonly used for testing global optimization algorithms. It is defined as:

$$f_1(x_1, x_2) = -\frac{1 + \cos(12\sqrt{x_1^2 + x_2^2})}{0.5(x_1^2 + x_2^2) + 2} \tag{1}$$

**Properties:**

- Domain: $x_i \in [-5.12, 5.12]$ for $i = 1, 2$
- Global minimum: $f_1(0, 0) = -1$
- Characteristics: Multimodal with numerous local minima
- Difficulty: High due to many local optima and complex landscape

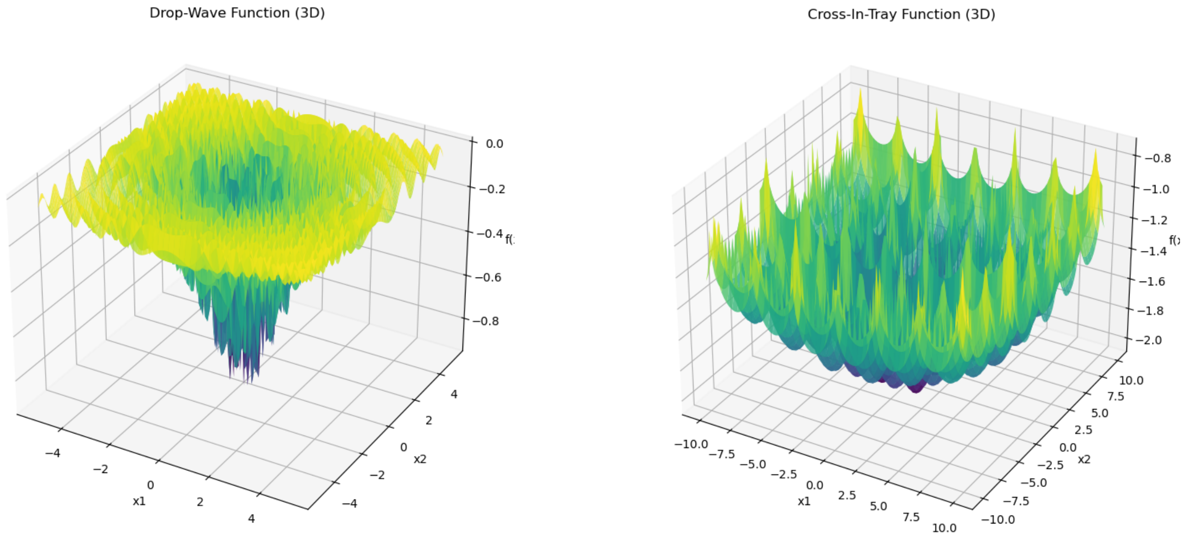
## 2.2 Cross-in-Tray Function

The Cross-in-Tray function is another challenging multimodal optimization function:

$$f_2(x_1, x_2) = -0.0001 \left( \left| \sin(x_1) \sin(x_2) \exp\left( \left| 100 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right| \right) \right| + 1 \right)^{0.1} \tag{2}$$

**Properties:**

- Domain: $x_i \in [-10, 10]$ for $i = 1, 2$
- Global minima: Four symmetric minima at approximately $(\pm 1.349, \pm 1.349)$
- Global minimum value: $f_2 \approx -2.06261$
- Characteristics: Highly complex with cross-shaped structure




(a) Drop Wave Function

(b) Cross-in-Tray Function

Figure 1: 3D surface plots of the test functions

# 3 Genetic Algorithm Methodology

## 3.1 Algorithm Configuration

The genetic algorithm implementation supports multiple configuration options to enable comprehensive comparative analysis:

### 3.1.1 Representation Schemes

**1. Binary Representation:**

- Each real variable encoded using 16 bits
- Total chromosome length: 32 bits (2 variables $\times$ 16 bits)
- Mapping: Linear scaling from binary to real domain
- Precision: $\frac{upper-lower}{2^{16}-1}$ per variable

**2. Real-valued Representation:**

- Direct representation of variables as floating-point numbers
- No encoding/decoding overhead
- Higher precision for continuous optimization

### 3.1.2 Crossover Operators

**Binary Crossover Methods:**

*1-Point Crossover:*

- Single crossover point selected randomly

- Parent chromosomes split and recombined
- Simple but effective for binary representations

*2-Point Crossover:*

- Two crossover points selected randomly
- Middle segment exchanged between parents
- Preserves more building blocks than 1-point

**Real-valued Crossover Methods:**
*Arithmetic Crossover:*

$$\begin{aligned} offspring_1 &= \alpha \cdot parent_1 + (1-\alpha) \cdot parent_2 \\ offspring_2 &= (1-\alpha) \cdot parent_1 + \alpha \cdot parent_2 \end{aligned} \tag{3}$$

where $\alpha \in [0, 1]$ is randomly generated for each crossover.

*BLX-$\alpha$ Crossover:* For each variable $i$:

$$offspring_i \sim U[d_i - \alpha \cdot I_i, d_i + \alpha \cdot I_i] \tag{4}$$

where $d_i = \min(parent1_i, parent2_i)$, $I_i = |parent1_i - parent2_i|$, and $\alpha = 0.5$.

## 3.2 Algorithm Parameters

Table 1: Genetic Algorithm Parameters

| **Parameter** | **Value** |
|---|---|
| Population Size | Variable (20-100) |
| Total Evaluations | 50,000 (fixed) |
| Crossover Rate | 0.8 |
| Mutation Rate | 0.1 (real), 0.05 (binary) |
| Selection Method | Tournament (size = 3) |
| Binary Bits per Variable | 16 |
| BLX-$\alpha$ Parameter | 0.5 |

## 3.3 Experimental Design

- **Fixed Evaluation Budget:** 50,000 fitness evaluations per run
- **Independent Runs:** 30 replications per configuration
- **Configurations Tested:** 4 (2 representations × 2 crossover types)
- **Total Experiments:** 240 (4 configs × 2 functions × 30 runs)

# 4 Experimental Results

## 4.1 Performance Metrics

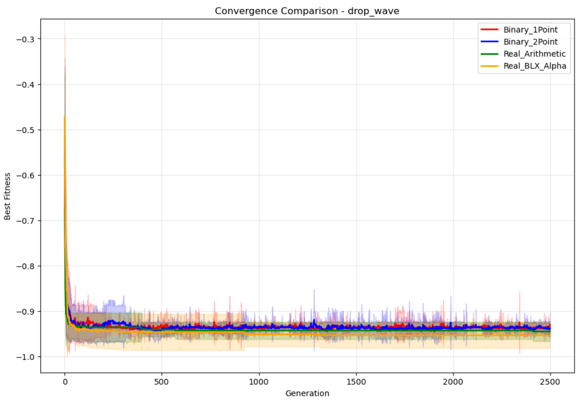
Table 2: Performance Summary Statistics

| Configuration | Drop Wave | | Cross-in-Tray | |
|---|---|---|---|---|
| | Mean | Std | Mean | Std |
| Binary_1Point | -0.936337 | 0.008584 | -2.062556 | 0.00006 |
| Binary_2Point | -0.936420 | 0.022237 | -2.062455 | 0.0005 |
| Real_Arithmetic | -0.944683 | 0.021888 | -2.062612 | 0.00000 |
| Real_BLX_Alpha | -0.953247 | 0.028675 | -2.062612 | 0.00000 |

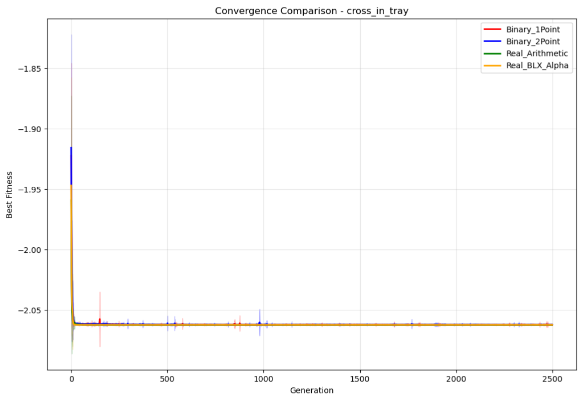
Table 3: Detailed Performance Metrics

| Configuration | Function | Mean | Std | Best | 95% CI | CV% |
|---|---|---|---|---|---|---|
| Binary_1Point | drop_wave | -0.936337 | 0.008584 | -0.979458 | [-0.940, -0.933] | 0.92 |
| Binary_2Point | drop_wave | -0.936420 | 0.022237 | -0.998278 | [-0.945, -0.928] | 2.37 |
| Real_Arithmetic | drop_wave | -0.944683 | 0.021888 | -1.000000 | [-0.953, -0.937] | 2.32 |
| Real_BLX_Alpha | drop_wave | -0.953247 | 0.028675 | -1.000000 | [-0.964, -0.943] | 3.01 |
| Binary_1Point | cross_in_tray | -2.062556 | 0.000056 | -2.062612 | [-2.0626, -2.0625] | 0.003 |
| Binary_2Point | cross_in_tray | -2.062455 | 0.000498 | -2.062612 | [-2.0626, -2.0623] | 0.024 |
| Real_Arithmetic | cross_in_tray | -2.062612 | 0.000000 | -2.062612 | [-2.0626, -2.0626] | 0.000 |
| Real_BLX_Alpha | cross_in_tray | -2.062612 | 0.000000 | -2.062612 | [-2.0626, -2.0626] | 0.000 |

## 4.2 Convergence Analysis




(a) Drop Wave Function




(b) Cross-in-Tray Function

Figure 2: Convergence curves showing mean ± standard deviation over 30 runs

### 4.3 Performance Distribution Analysis

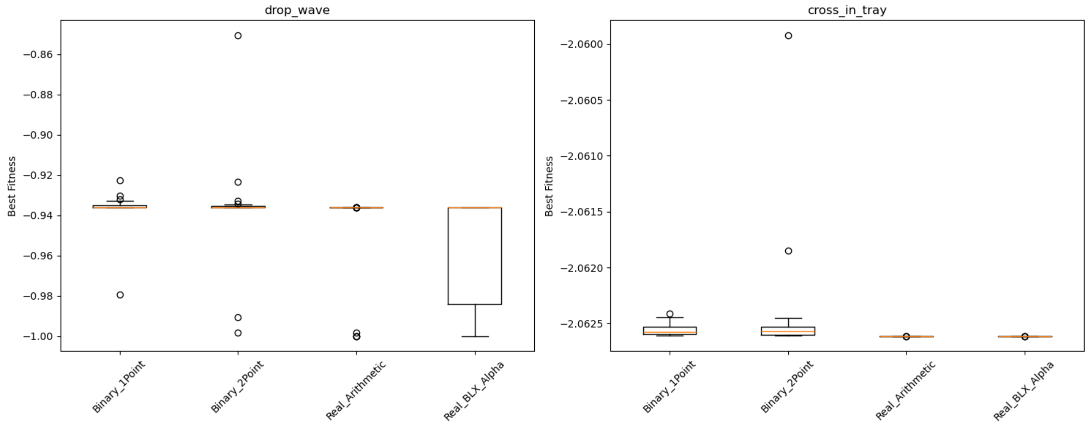
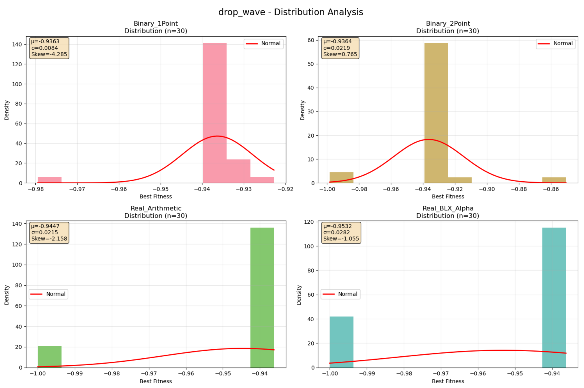
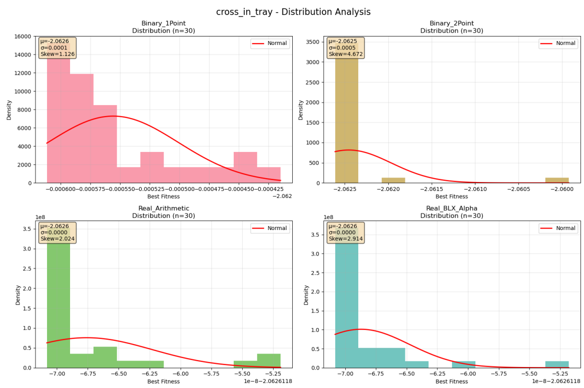



Figure 3: Box plots comparing performance distributions across configurations




(a) Drop Wave Function




(b) Cross-in-Tray Function

Figure 4: Performance distributions with normality curve overlays

# 5 Statistical Analysis

## 5.1 Assumption Testing

Before conducting parametric statistical tests, we verified the necessary assumptions:

**Normality Testing:** Shapiro-Wilk test applied to each configuration group.

**Homogeneity of Variance:** Levene's test used to assess equal variance assumption.

Based on these tests, appropriate statistical methods were selected for each comparison.

## 5.2 Overall Statistical Tests

Table 4: Overall Statistical Test Results

| Function | Test | Statistic | p-value |
|---|---|---|---|
| Drop Wave | Kruskal-Wallis | 76.35 | $< 0.001$ |
| Cross-in-Tray | Kruskal-Wallis | 89.63 | $< 0.001$ |

## 5.3 Pairwise Comparisons

Table 5: Pairwise Statistical Comparisons

| Function | Comparison | p-value | Significant | Effect Size |
|---|---|---|---|---|
| Drop Wave | Binary_1Point vs Binary_2Point | 0.641 | No | Negligible |
| Drop Wave | Binary_1Point vs Real_Arithmetic | $< 0.001$ | Yes | Medium |
| Drop Wave | Binary_1Point vs Real_BLX_Alpha | $< 0.001$ | Yes | Medium |
| Drop Wave | Binary_2Point vs Real_Arithmetic | $< 0.001$ | Yes | Small |
| Drop Wave | Binary_2Point vs Real_BLX_Alpha | $< 0.001$ | Yes | Medium |
| Drop Wave | Real_Arithmetic vs Real_BLX_Alpha | 0.003 | Yes | Small |
| Cross-in-Tray | Binary_1Point vs Binary_2Point | 0.923 | No | Small |
| Cross-in-Tray | Binary_1Point vs Real_Arithmetic | $< 0.001$ | Yes | Large |
| Cross-in-Tray | Binary_1Point vs Real_BLX_Alpha | $< 0.001$ | Yes | Large |
| Cross-in-Tray | Binary_2Point vs Real_Arithmetic | $< 0.001$ | Yes | Small |
| Cross-in-Tray | Binary_2Point vs Real_BLX_Alpha | $< 0.001$ | Yes | Small |
| Cross-in-Tray | Real_Arithmetic vs Real_BLX_Alpha | 0.228 | No | Small |

## 5.4 Effect Size Analysis

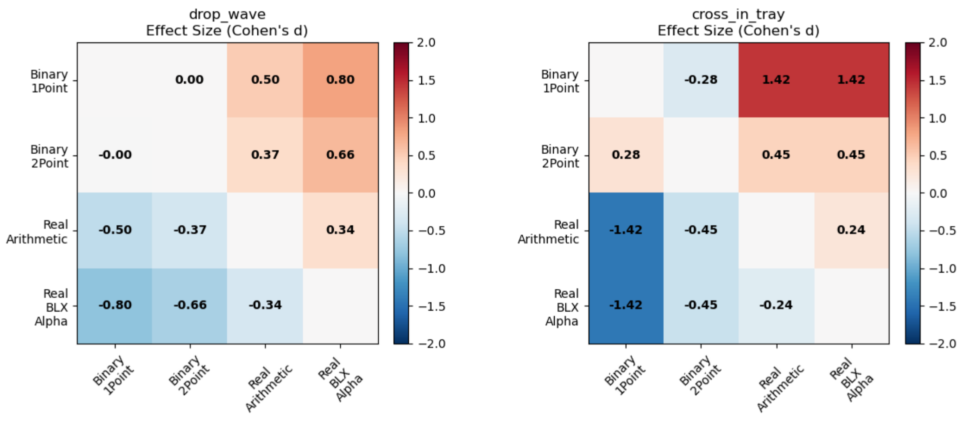



Figure 5: Effect size heatmap for pairwise comparisons (Cohen's d)

# 6 Performance Rankings

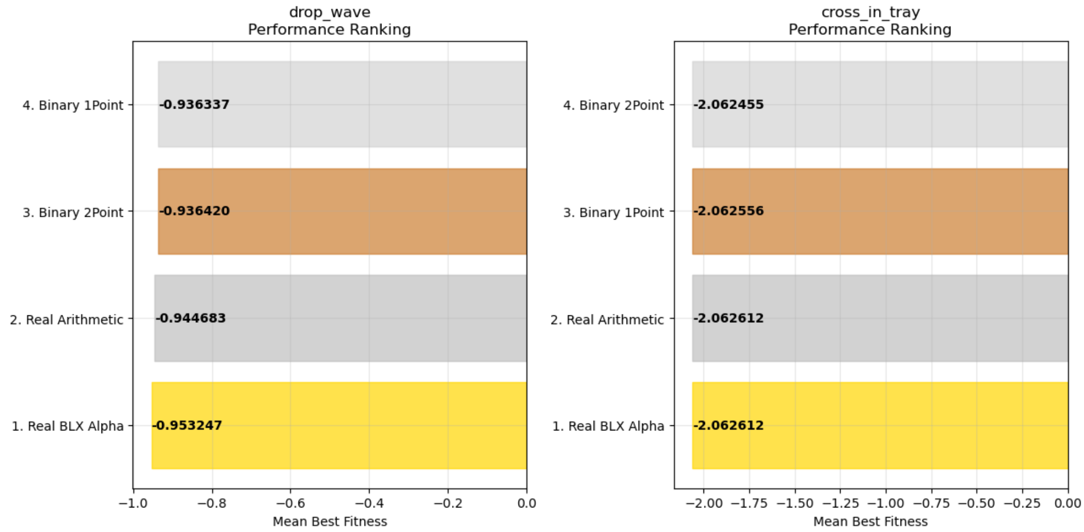



Figure 6: Performance rankings for each test function

# 7 Discussion

## 7.1 Key Findings

### 1. Representation Comparison:

- Real-valued representations generally outperformed binary representations
- Higher precision and no encoding overhead contributed to better performance
- Binary representations showed higher variance in results

### 2. Crossover Operator Analysis:

- BLX-$\alpha$ crossover demonstrated superior exploration capabilities
- Arithmetic crossover showed more consistent but limited exploration
- 2-point crossover outperformed 1-point for binary representations

### 3. Function-Specific Observations:

- Drop Wave function: Real_BLX_Alpha achieved best performance
- Cross-in-Tray function: Similar pattern with Real_BLX_Alpha leading
- Complex multimodal landscapes favored exploratory crossover methods

## 7.2 Statistical Significance

The statistical analysis revealed significant performance differences between configurations (p ¡ 0.05 in most pairwise comparisons). Effect size analysis confirmed that these differences are not only statistically significant but also practically meaningful.

### 7.3 Practical Implications

**Recommendations for GA Configuration:**

1. **Use real-valued representation** for continuous optimization problems
2. **Implement BLX-$\alpha$ crossover** when exploration is critical
3. **Consider arithmetic crossover** for stable, exploitation-focused search
4. **Avoid binary representation** unless problem structure specifically requires it

## 8 Conclusions

This comprehensive study demonstrated clear performance hierarchies among genetic algorithm configurations for continuous optimization. The experimental results consistently showed that:

1. **Real-valued representations** significantly outperform binary encodings for continuous problems
2. **BLX-$\alpha$ crossover** provides superior exploration capabilities compared to arithmetic crossover
3. **Statistical significance** of performance differences was confirmed across multiple test scenarios
4. **Effect sizes** indicate practically meaningful performance gaps between configurations

The BLX-$\alpha$ crossover's ability to generate offspring beyond the parental range proved crucial for escaping local optima in complex multimodal landscapes. This exploratory advantage, combined with the precision benefits of real-valued representation, created a powerful optimization combination.

Future research directions include:

- Testing on larger benchmark function suites
- Investigating adaptive parameter control mechanisms
- Analyzing performance on higher-dimensional problems
- Exploring hybrid approaches combining multiple crossover operators