Optimized Genetic Algorithms in Minimization

Georgiana Diana Ojoc

December 3, 2019

Abstract

This paper presents how a Genetic Algorithm can be improved using Adaptive Crossover, Adaptive Mutation and the Tournament Selection technique, instead of the Roulette Wheel Technique, which was described in the previous article. Also, it was found that applying the Gray Code did not improve the results.

1 Introduction and Motivation

The goal of a **Genetic Algorithm** is to come as close as possible to the optimal solution. Since the solution search space is so huge, the major difficulty in reaching this goal is the convergence into local minima before exploring the entire search space for global minima. This is where the concept of adaptive operators can be exploited to help add some order in the random search for near optimal solution.

2 Method

Mutation

For each generation, a ratio between average fitness and maximum fitness was calculated, in order to determine if the algorithm converges into a local minimum. Thus, if this ratio is greater than **0.9**, the probability of mutation is increased by **0.0002**, and if this ratio is lower than **0.7**, the probability of mutation is decreased by **0.0001**.

Crossover

If the previously calculated ratio is greater than **0.9**, the Single Point Crossover, in order to preserve the good gene segments from the parents effectively. Otherwise, the 5-Point Crossover is applied, wherein alternating segments are swapped to get new off-springs.
Another way to improve the Crossover is to prevent the incest between two parents, which means that two chromosomes with a Hamming Distance lower than the number of dimensions will not swap gene segments.

Selection

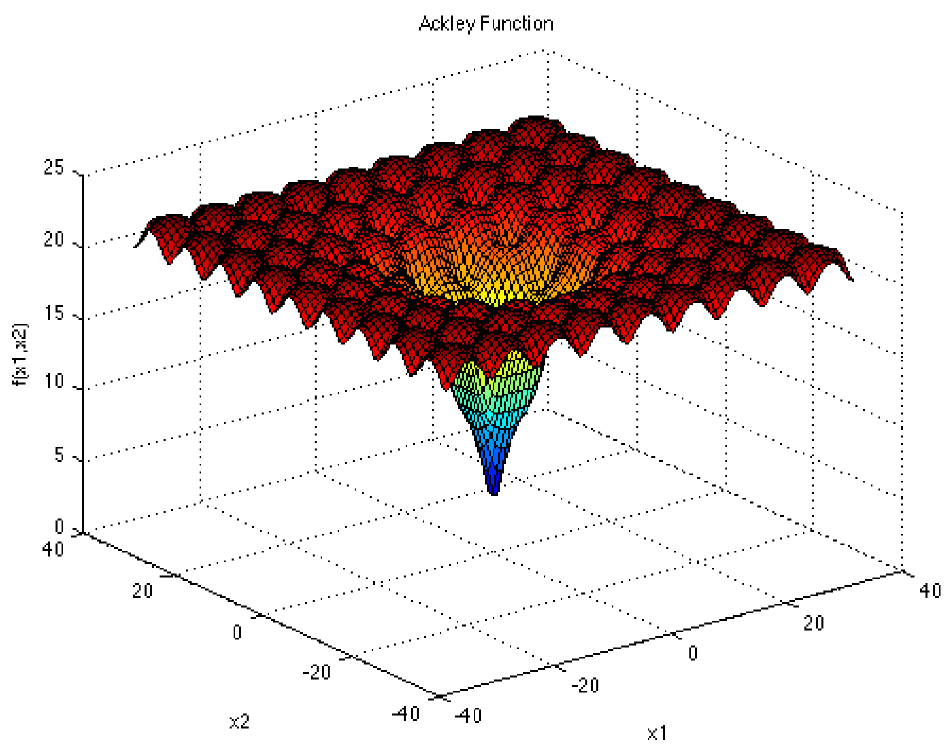
The Roulette Wheel technique was replaced with the 20-Way Tournament Selection technique, which selects 20 individuals and run a tournament among them. Only the fittest candidate amongst those selected candidates is chosen and is passed on to the next generation. If the tournament size is larger, weak candidates have a smaller chance of getting selected as they have to compete with stronger candidates. Tournament Selection also works for negative fitness values.

3 Results

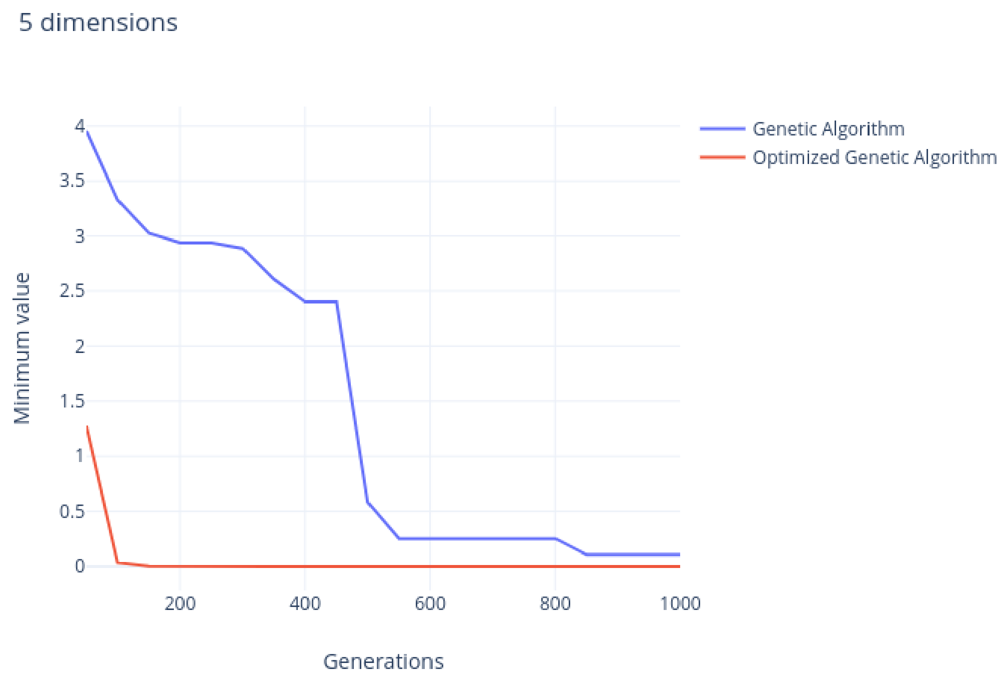
Ackley Function

Ackley's function is a widely used multimodal test function.

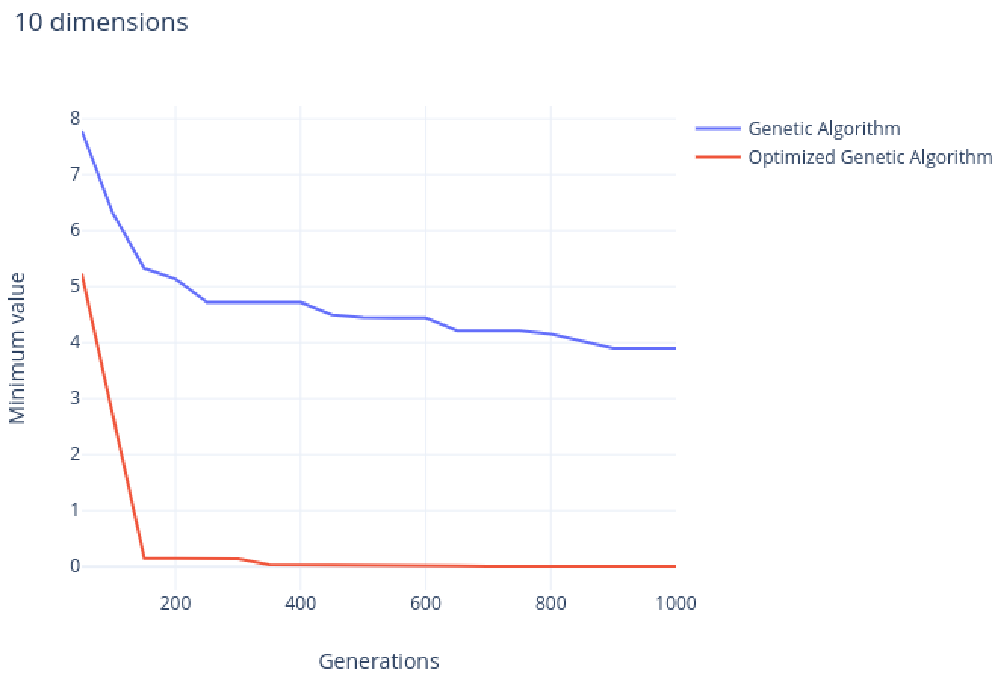
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)}$
- Search domain: $-15 \leqslant x_i \leqslant 30,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (0, 0, ..., 0),\ f(x^*) = 0$
- Function graph: for $n = 2$



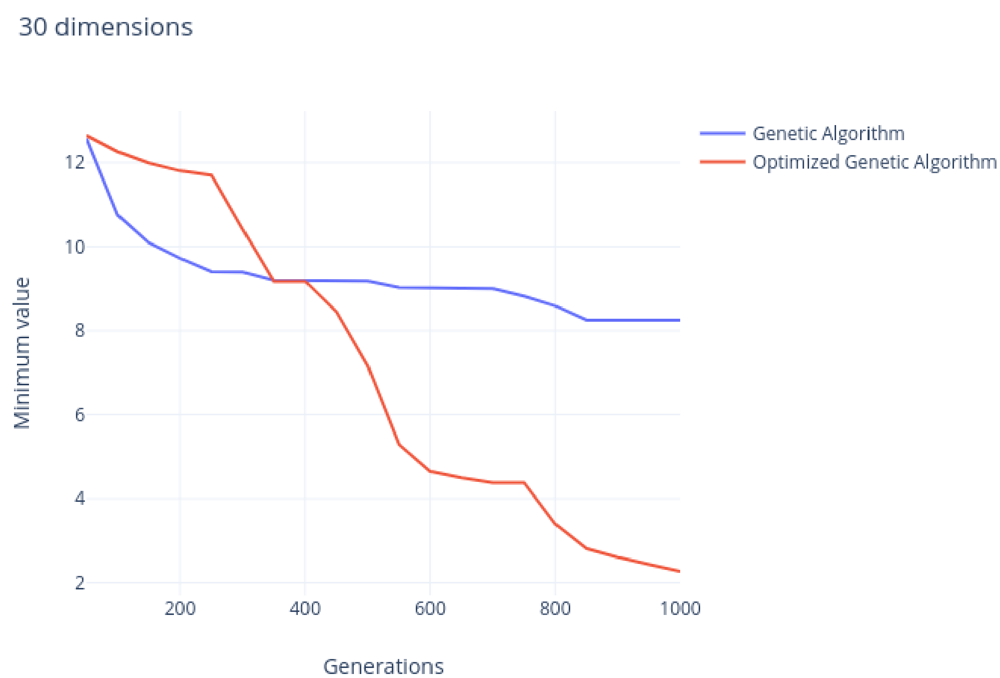
5 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	0.105677	0.000012
Highest value	3.528669	0.465890
Mean	2.462089	0.049494
Median	2.585520	0.000053
Standard deviation	1.027170	0.136212



10 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	3.896640	0.004862
Highest value	8.767346	1.830841
Mean	7.203360	0.868248
Median	7.950652	0.847527
Standard deviation	1.656964	0.862092



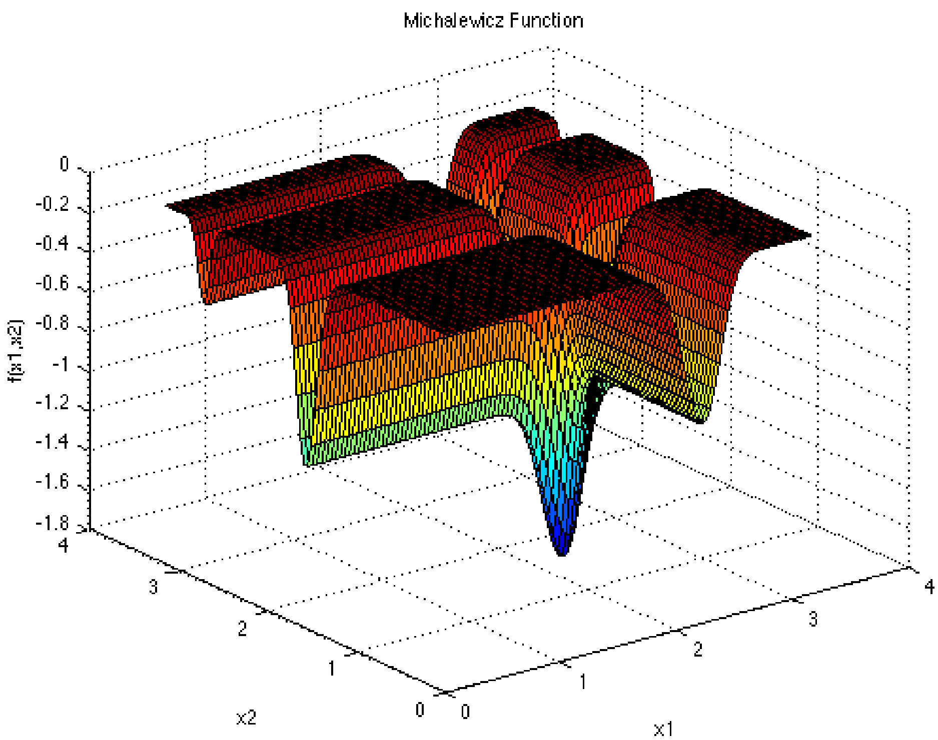
30 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	8.251607	2.271223
Highest value	9.885168	4.814041
Mean	9.059375	3.781378
Median	9.055507	3.790205
Standard deviation	0.474584	0.676339



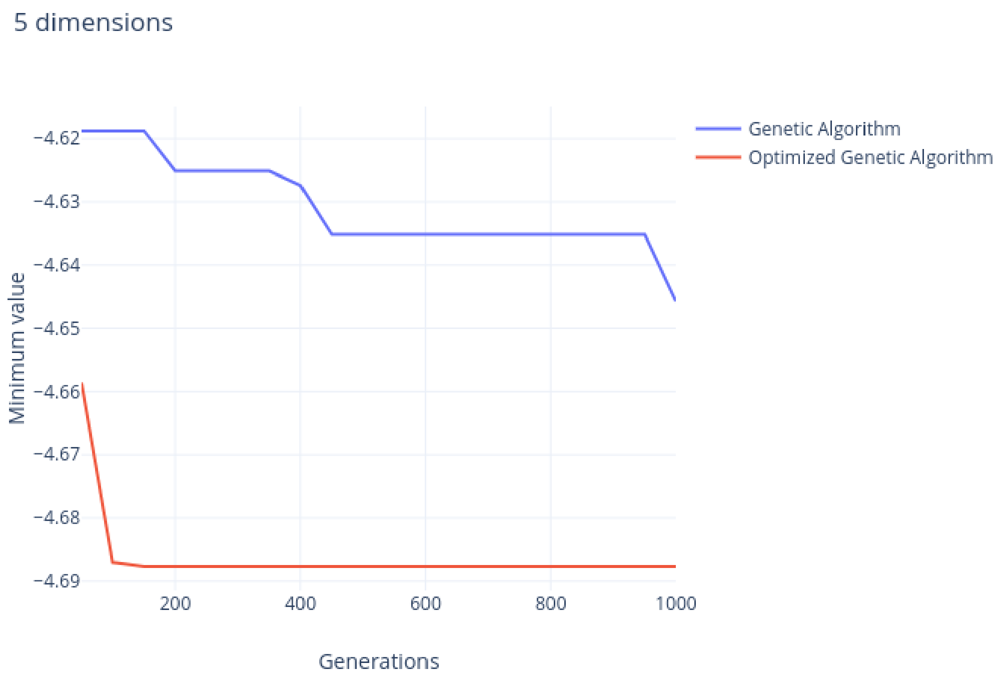
Michalewics Function

Michalewics' function is a multimodal test function. The exponent of the second sinus defines the "steepness" of the valleys or edges. Larger exponent leads to more difficult search. For very large exponent the function behaves like a needle in the haystack (the function values for points in the space outside the narrow peaks give very little information on the location of the global optimum).

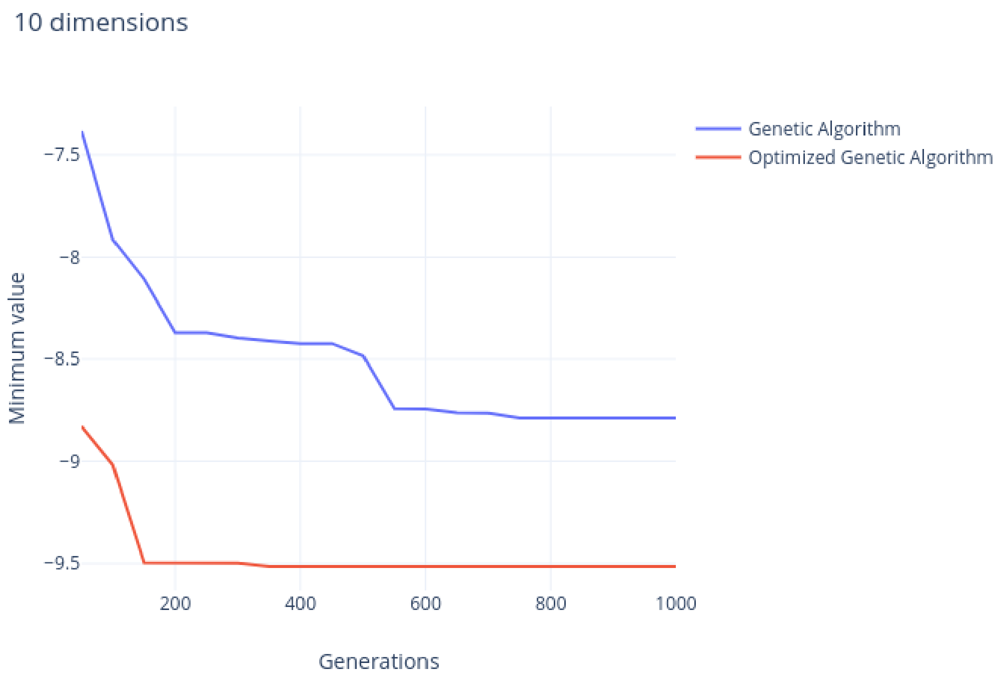
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = -\sum_{i=1}^{n} \sin(x_i) \sin^{20}(\frac{ix_i^2}{\pi})$
- Search domain: $0 \leqslant x_i \leqslant \pi,\ i = 1, 2, ..., n$
- Global minimum: $n = 5 : f(x^*) = -4.687658$, $n = 10 : f(x^*) = -9.660150$
- Function graph: for $n = 2$



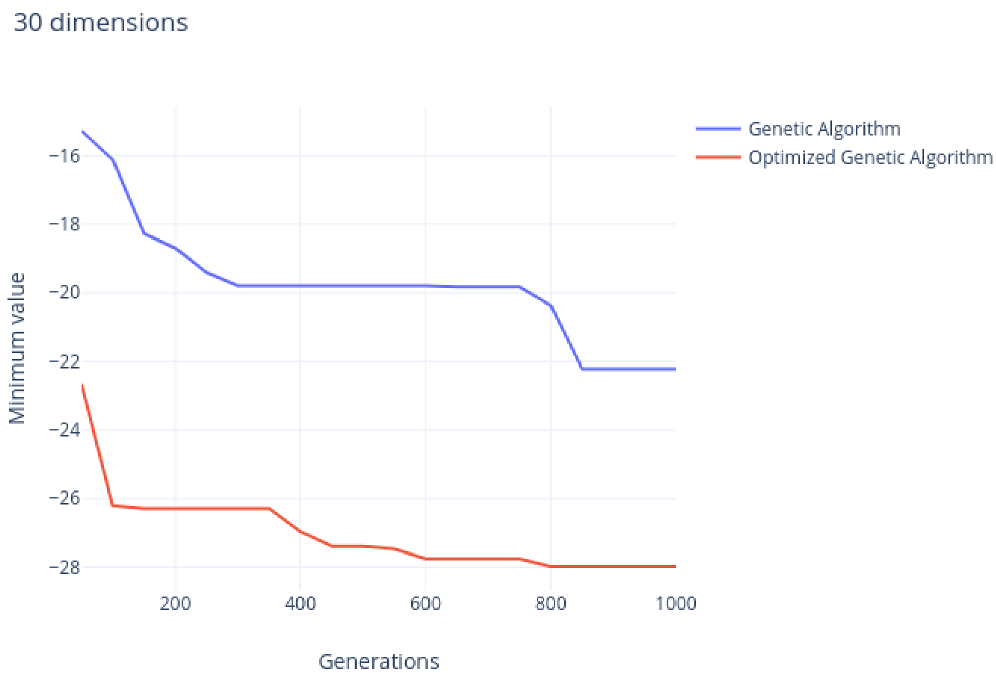
5 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	-4.645770	-4.687658
Highest value	-4.352910	-4.374896
Mean	-4.522150	-4.614195
Median	-4.520190	-4.652186
Standard deviation	0.111128	0.095901



10 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	-8.877780	-9.514810
Highest value	-8.036360	-8.971087
Mean	-8.574890	-9.282084
Median	-8.565210	-9.260985
Standard deviation	0.262389	0.153140



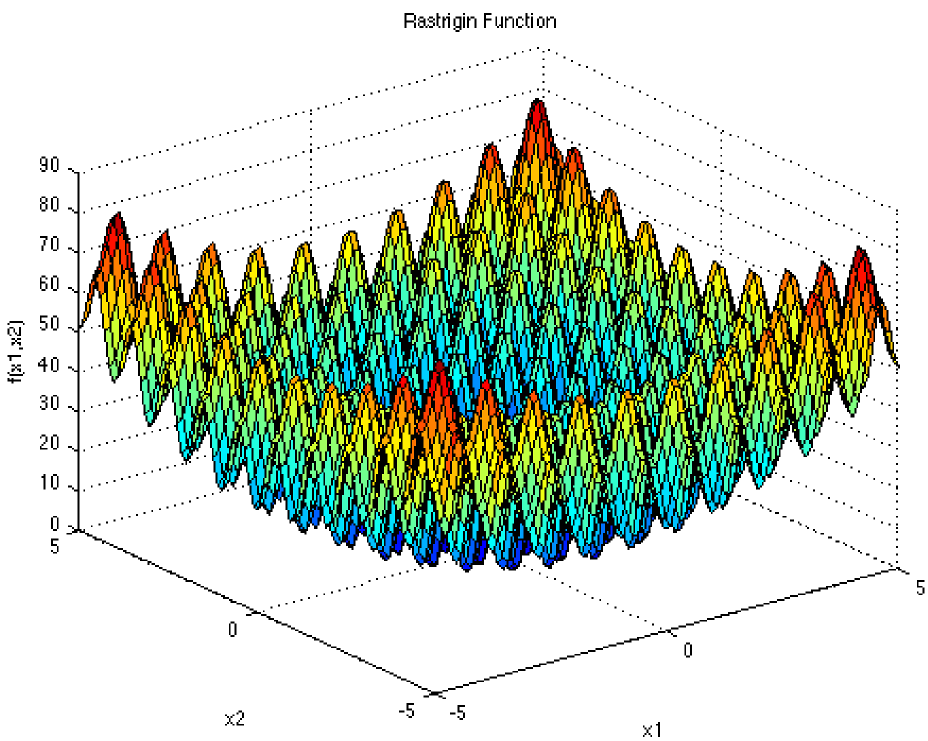
30 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	-22.218564	-27.980700
Highest value	-20.042127	-26.273900
Mean	-21.483559	-26.878100
Median	-21.716687	-26.931500
Standard deviation	0.645083	0.433566



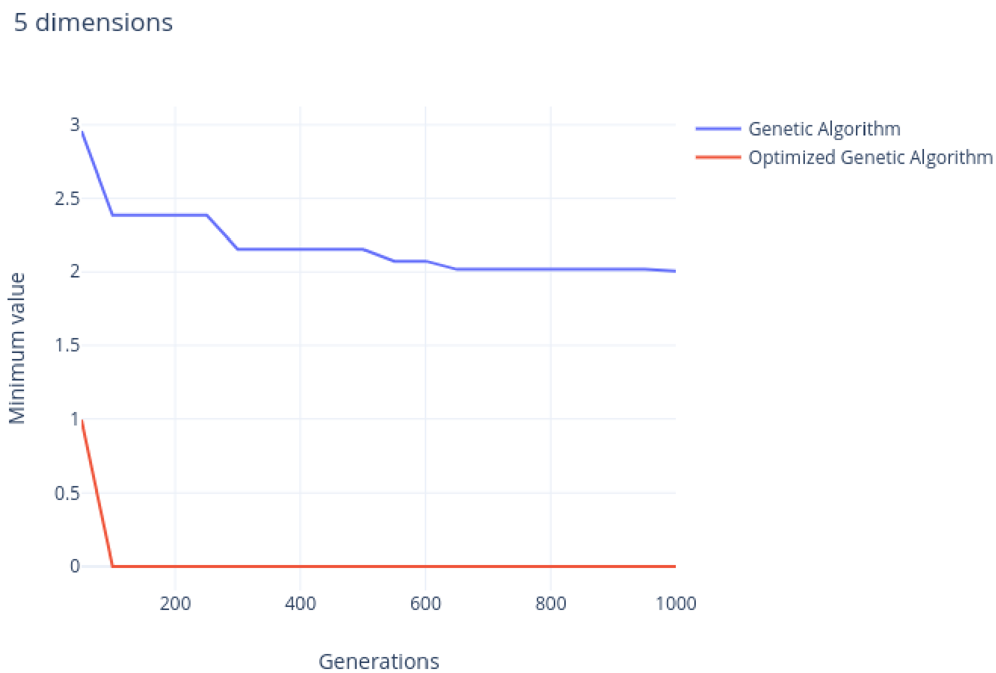
Rastrigin Function

Rastrigin's function produces many local minima. Thus, the test function is highly multimodal. However, the location of the minima are regularly distributed.

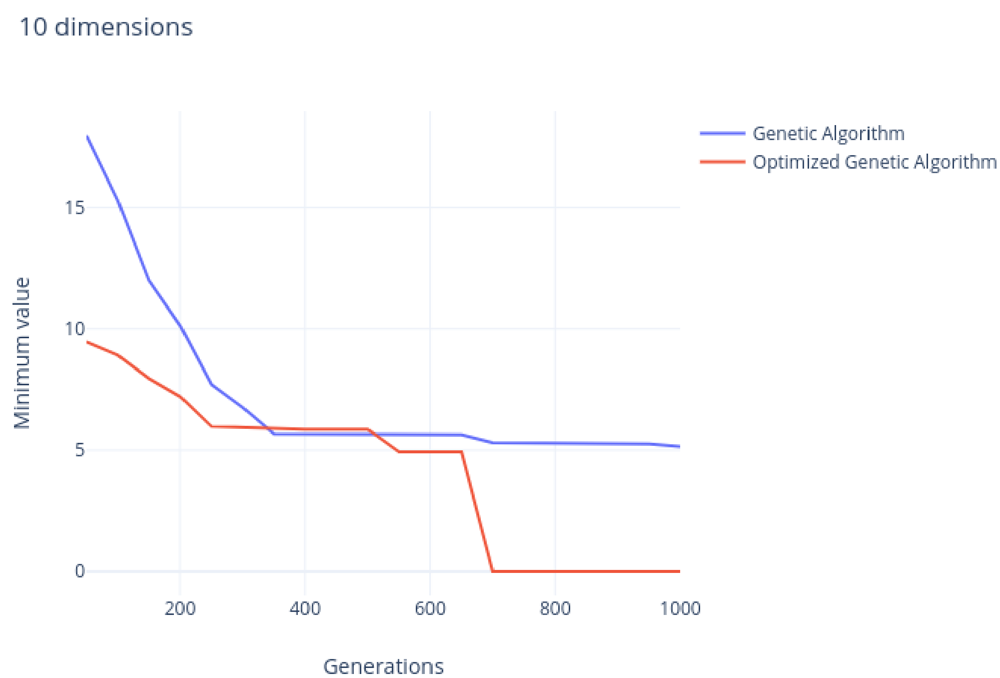
- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = 10n + \sum_{i=1}^{n}(x_i^2 - 10cos(2\pi x_i))$
- Search domain: $-5.12 \leqslant x_i \leqslant 5.12,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (0, 0, ..., 0),\ f(x^*) = 0$
- Function graph: for $n = 2$



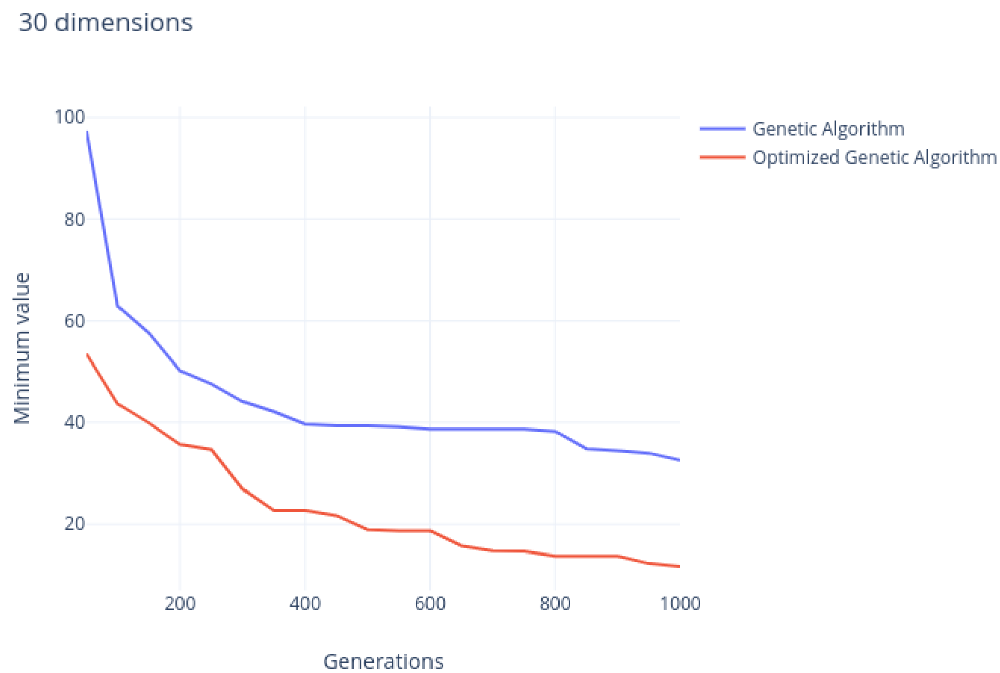
5 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	2.004868	0
Highest value	3.316668	1.471540
Mean	2.539986	0.230765
Median	2.311425	0.235768
Standard deviation	0.476529	0.359128



10 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	5.152758	0.000213
Highest value	8.671650	3.714068
Mean	7.139388	2.048768
Median	7.617864	1.581611
Standard deviation	1.447347	1.324375



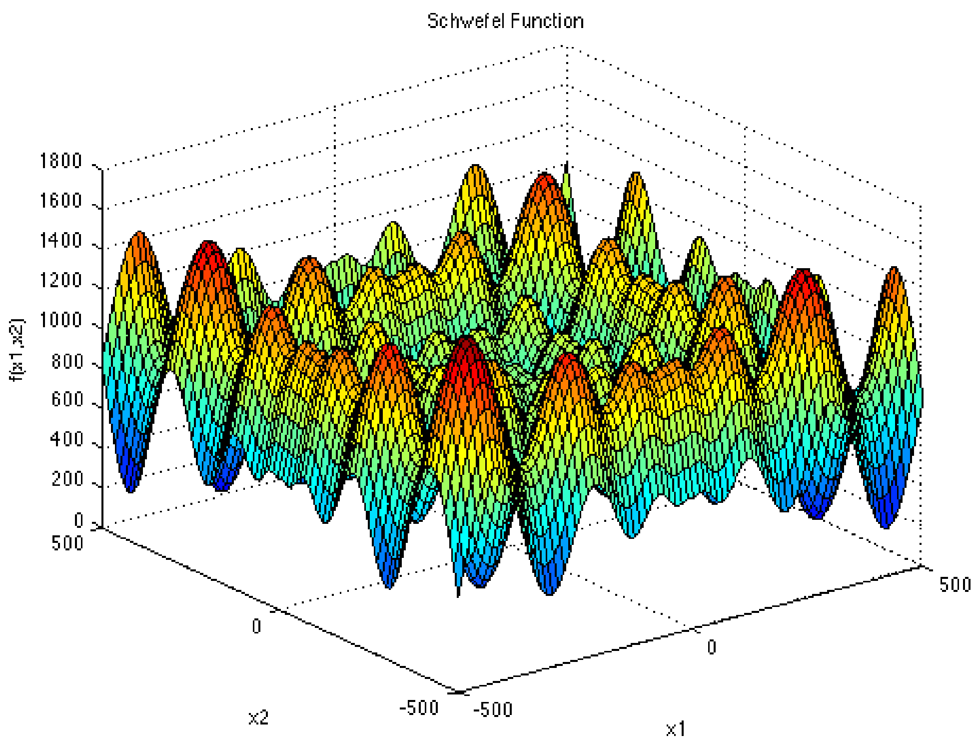
30 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	32.608656	11.730428
Highest value	47.659114	29.448395
Mean	39.260717	23.593918
Median	37.573377	24.641484
Standard deviation	5.696159	5.663869



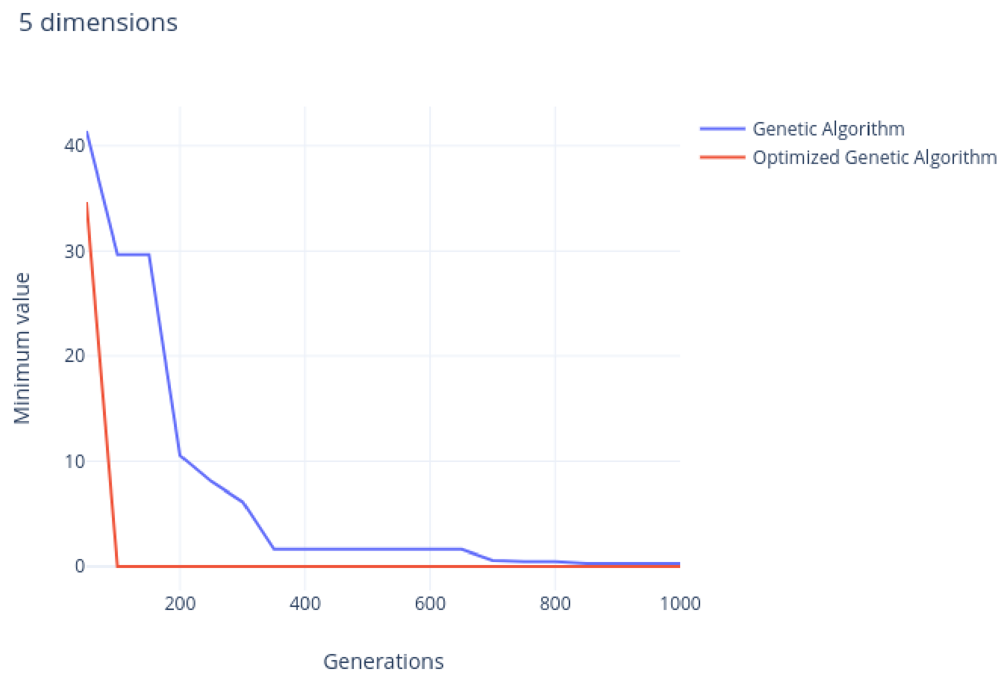
Schwefel Function

Schwefel's function is deceptive in that the global minimum is geometrically distant, over the parameter space, from the next best local minimum. Therefore, the search algorithms are potentially prone to convergence in the wrong direction.

- Number of variables: n variables
- Definition: $f(\mathbf{x}) = f(x_1, ..., x_n) = 418.9829n - \sum_{i=1}^{n} x_i \sin(\sqrt{|x_i|})$
- Search domain: $-500 \leqslant x_i \leqslant 500,\ i = 1, 2, ..., n$
- Global minimum: $x^* = (420.9687, 420.9687, ..., 420.9687),\ f(x^*) = 0$
- Function graph: for $n = 2$

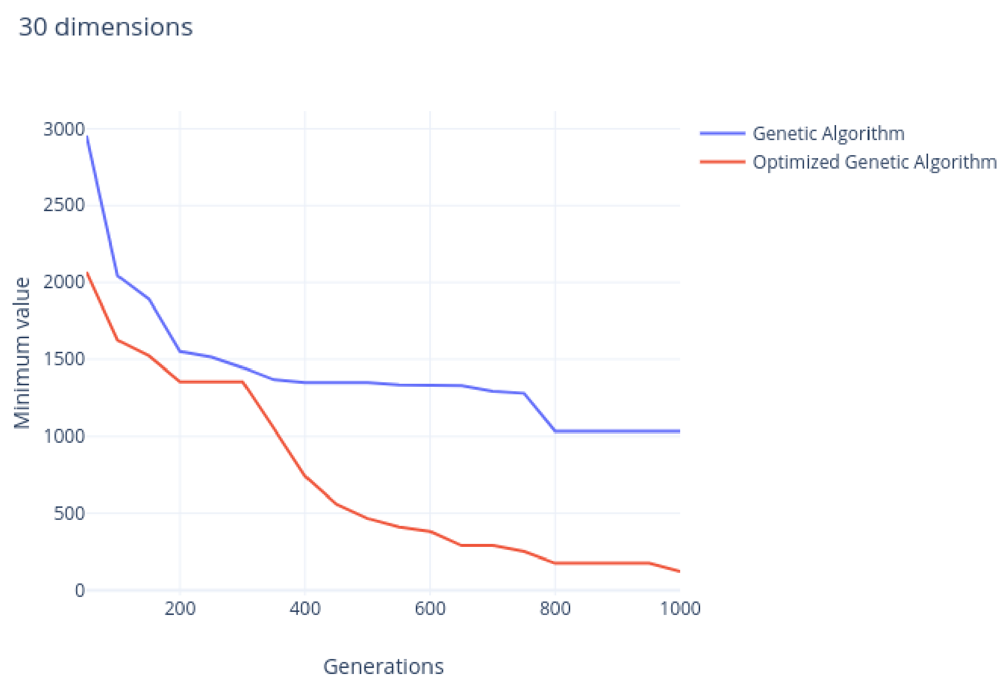


5 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	0.280626	0.000688
Highest value	172.860100	0.518478
Mean	69.210550	0.266052
Median	51.890990	0.311073
Standard deviation	67.686830	0.141339



10 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	154.773460	0.425084
Highest value	354.522116	0.757278
Mean	284.524196	0.563806
Median	306.180088	0.541152
Standard deviation	61.859351	0.094269

30 dimensions *Minimum value*	**Genetic Algorithm**	**Optimized Genetic Algorithm**
Lowest value	1033.183208	123.341435
Highest value	1813.277007	479.740864
Mean	1525.734830	332.476763
Median	1548.452866	353.947031
Standard deviation	246.819969	112.392925



4 Interpretation and Conclusion

The Genetic Algorithm was improved by keeping around **0.85** the ratio between the average fitness and the maximum fitness for each generation. Also, the conversion from Gray Code to binary was not useful, because the minimum value of the Rastrigin function for 30 dimensions was greater than 100 in this circumstances.

References

- https://pdfs.semanticscholar.org/0bcf/8157a7a35ecb9b6836902f65a362e5db2552.pdf
- https://www.sfu.ca/~ssurjano/ackley.html
- https://www.sfu.ca/~ssurjano/michal.html
- https://www.sfu.ca/~ssurjano/rastr.html
- https://www.sfu.ca/~ssurjano/schwef.html