

Adaptive Evolutionary Computation

on Numerical Optimization

Isaac Lam

Institute of Data Science and Engineering
National Yang Ming Chiao Tung University

6-Feb, 2024

Summary

- 1 Introduction
- 2 Related Work
- 3 Method
- 4 Result
- 5 Discussion

Introduction

Evolutionary algorithms

- Inspired by natural evolution and genetic principles
- Involves populations of candidate solutions (individuals)
- Key operations: selection, crossover, mutation
- Goal: Optimize solutions for complex problems
- Widely used in optimization, machine learning, and AI

Challenges of Standard EC

- **Premature Convergence:** Without adaptation, EC methods can converge too early to suboptimal solutions.
- **Lack of Robustness:** Fixed parameters (e.g. mutation rates, crossover rates) may not work well across different types of problem.
- **Inefficiency:** In dynamic environments, EC can be inefficient if the same parameters are used for all generations.

Adaptive Evolutionary Computation

Algorithm can modify its own parameters during the evolutionary process.

- Self-Adaptive

- Parameters encoded directly within the genotype.
- evolve alongside with candidate solutions

- Adaptive

- Controlled externally by rules or feedback from the evolution.
- 1/5 Success Rule

Related Work

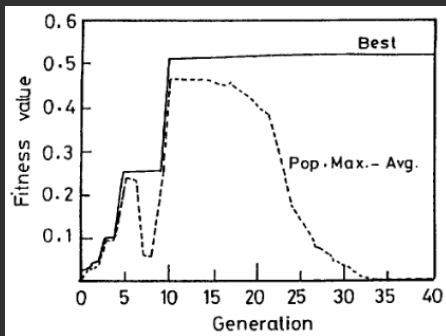
1/5 Success Rule

Adjust the mutation rate based on the success of recent generations.

- If success rate $< 1/5$, increase mutation rate (exploration).
- If success rate $> 1/5$, decrease mutation rate (exploitation).
- Otherwise, mutation rate remain unchanged.

Fitness Based

- Change the mutation rate with current fitness and the average fitness by the ratio between them. [7]
- A local minima can be detected by the difference between population maximum fitness and average fitness. [14]



Diversity Based

- Monitoring and Adjustment
 - Measuring the population diversity using metrics like average distance[5, 9] or Shannon Entropy[15] .
 - If diversity drops, increase mutation rate to promote exploration and avoid stagnation.
- Distance-Based Crossover Control
 - Often refer to Mating Distance [2].
 - Enforcing distance between parents.
 - By ensure the genetic difference to maintain population diversity.

Method

Adaptive Method

- 1/5 Success Rule
- Average Fitness
- Generations
- Diversity
- Cosine Similarity
- Mating Distance

Average Fitness

■ Individual-Based

- For each individual, if its fitness is lower than the average fitness of the population, increase the mutation rate.
- This encourages exploration for individuals that are not performing well compared to the population.

■ Global-Based

- Compare the current average fitness with the previous average fitness of the population.
- If the average fitness stagnates or doesn't improve, increase mutation rate to reintroduce diversity and avoid local minima.

Generations

- Adjust the mutation step size according to difference phase of the searching process.
- Inspired from the scheduler of gradient descent.
- A larger mutation step size is applied at the beginning to increase exploration.
- Progressively lower the mutation step size to increase the exploitation.
- One typical approach is Cosine Annealing. [6]

$$\sigma_t = \sigma_{min} + \frac{1}{2}(\sigma_{max} - \sigma_{min}) \left(1 + \cos \left(\frac{T_{cur}}{T_{max}} \pi \right) \right)$$

Diversity - Distance Function

■ Jaccard Distance [4]

$$d_J(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

■ Euclidean Distance

$$d_J(X_i, X_j) = \sqrt{\sum_{k=1}^n (x_{ik} - x_{jk})^2}$$

■ Population Diversity

$$\text{Average Pairwise Distance} = \frac{1}{N(N-1)} \sum_{i \neq j} d(X_i, X_j)$$

Distance - Drawback

■ Unbounded Range

- The average pairwise distance can grow unbounded, making it difficult to set a fixed threshold for diversity.
- The threshold would need to be problem-specific, and setting a good value is challenging.
- Can be solved by compare the current diversity with previous generations' results to adaptively set thresholds.

■ Curse of Dimensionality

- In higher-dimensional spaces, distances become less meaningful as the points become sparsely distributed.
- Reduces the effectiveness of the diversity measure in large search spaces.
- A well-known example is hubness problem in kNN. [10]

Cosine Similarity

- bounded Range $[0, 1]$
 - Unlike Euclidean distance, Cosine Similarity is bounded within the range $[0, 1]$, making it easier to define thresholds for diversity.
 - This ensures a consistent scale across different problems and easier interpretation.
- Solving the Curse of Dimensionality
 - Since cosine similarity measures the angle between vectors, it is less sensitive to dimensionality and the sparsity of data.
 - In high-dimensional spaces, Euclidean distance can become less meaningful, but cosine similarity remains effective.

$$\text{Cosine Similarity}(X_i, X_j) = \frac{X_i \cdot X_j}{\|X_i\| \|X_j\|}$$

Where \cdot is dot product, and $\|X\|$ is vector norm.

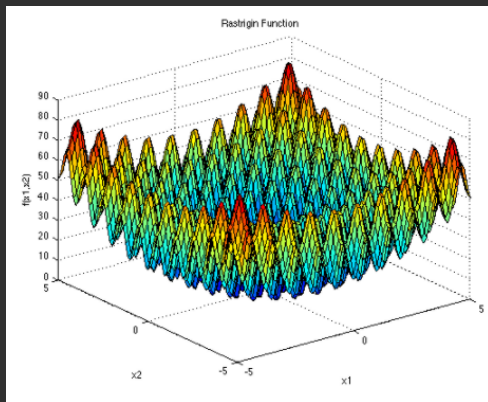
Mating Distance

- Select parents for crossover only if their genetic distance exceed certain threshold.
- Balancing mating distance helps maintain diversity and avoid premature convergence

Rastrigin Function

A non-convex function that contains a lot of local minima. [11, 13]

$$f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)]$$



Rosenbrock Function

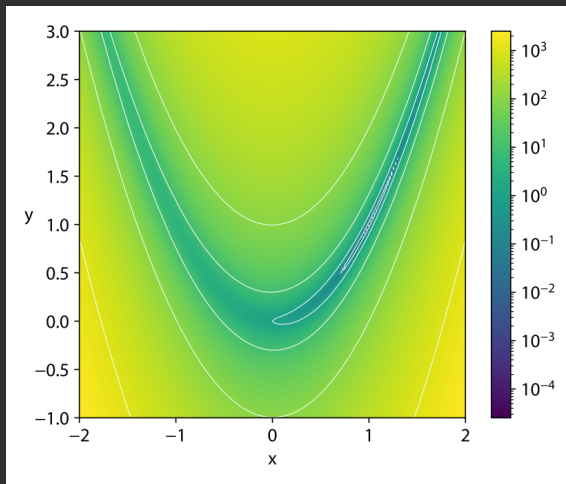
Rosenbrock Function [12] is much smoother than Rastrigin Function. The local minima are in a very narrow valley. There are two forms of the generalized function. The first one only defined for even N [1].

$$f(\mathbf{x}) = \sum_{i=1}^{N/2} [100(x_{2i-1}^2 - x_{2i})^2 + (x_{2i-1} - 1)^2]$$

The second one is more generalized and contains multiple minima and saddle point [3]:

$$f(\mathbf{x}) = \sum_{i=1}^{N-1} [100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2]$$

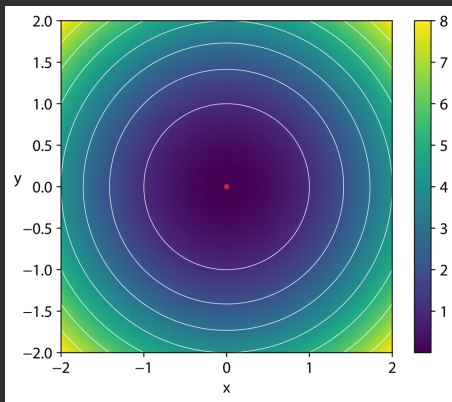
Rosenbrock Function



Sphere Function

A smooth convex function.

$$f(x) = \sum_{i=1}^n x_i^2$$



Experiment Setup

- Population Size: 10
- Offspring Size: 50
- Parent Selection: Fitness Proportion Selection
- Offspring Selection: Elitism without Replacement
- Crossover Operator: Discrete Recombination
- Mutation Operator: Uniform Mutation
- Initial Mutation Step Size: 0.05
- Maximum Generation: 100000
- Observe Window: 20
- Implementation: C++

More details can be found at [isaackhlam/Evolutionary-Computing](https://isaackhlam.com/Evolutionary-Computing)

Result

Average Generations

	Rastrigin	Rosenbrock	Sphere
No Adoption	9836.5	325752.6	1455.8
1/5 Success Rule	10037.7	292861.4	1592.8
Average Distance	9634.6	289984.7	1555.4
Similarity Compare	9662.2	291085.4	1490.1
Similarity Cutoff	10073.5	307112.0	1443.0
Cosine Annealing	9631.7	276111.3	1586.3
Average Fitness	9676.2	462220.5	1543.9
Mating Distance	9951.9	473526.2	2046.8

Statistical Test

We can't just eyeball to check whether these method improved. Instead, a statistical Test should be use. Since the number of run is small ($n = 10$), we can't assume normality and the Welch's t-test[16] may not be accurate enough.

Mann-Whitney U Test[8] can be used in this case.

$$U_i = R_i - \frac{n_i(n_i + 1)}{2}$$

$$U = \min(U_i)$$

$$z = \frac{U - \mu_U}{\sigma_U}, \text{ where } \mu_U = \frac{n_1 n_2}{2}, \sigma_U = \sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}}$$

$$N_0 : \mu_1 \leq \mu_2, N_a : \mu_1 > \mu_2$$

Mann-Whitney U Test - p-value

	Rastrigin	Rosenbrock	Sphere
1/5 Success Rule	0.83	0.43	0.79
Average Distance	0.43	0.48	0.66
Similarity Compare	0.63	0.55	0.32
Similarity Cutoff	0.71	0.63	0.43
Cosine Annealing	0.34	0.43	0.81
Average Fitness	0.45	0.96	0.57
Mating Distance	0.69	0.94	1.00

Discussion

Why not significant

- Rapid adjustment of the mutation rate lead to fluctuation.
- The adaption scheme quickly hit the mutation rate bound, making them indifferent.
- The operation and other parameters not varying may not suit these function.
- The number of experiment is small so the statistical test may not reliable.
- There may be potential bug or wrong implementation I didn't notice.

Limitation and Future Work

■ Limitation

- Only tested on Non-constraints problem.
- Only tested with a few parameters. (scale ratio, observe window, etc.)
- Only tested with fixed crossover and mutation operator.

■ Future Work

- Test on constraint optimization problem like knapsack problem.
- Conduct a comprehensive parameter search on threshold, scale values.
- Discuss whether the extra computation worth.

References I

1. Dixon, L. C. W. & Mills, D. J. Effect of rounding errors on the variable metric method. en. *J. Optim. Theory Appl.* **80**, 175–179 (Jan. 1994).
2. Eshelman, L. J. *Preventing premature convergence in genetic algorithms by preventing incest.* in *Proceedings of Fourth International Conference on Genetic Algorithms, 1991* (1991).
3. Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning.* 1st. ISBN: 0201157675 (Addison-Wesley Longman Publishing Co., Inc., USA, 1989).
4. Jaccard, P. Nouvelles Recherches Sur la Distribution Florale. *Bulletin de la Societe Vaudoise des Sciences Naturelles* **44**, 223–70 (Jan. 1908).

References II

5. Jia, Y. *et al.* Unravelling fish community assembly in shallow lakes: Insights from functional and phylogenetic diversity. *Reviews in Fish Biology and Fisheries* **32** (Oct. 2021).
6. Loshchilov, I. & Hutter, F. *SGDR: Stochastic Gradient Descent with Warm Restarts*. 2017. arXiv: 1608.03983 [cs.LG].
<https://arxiv.org/abs/1608.03983>.
7. Marsili Libelli, S. & Alba, P. Adaptive mutation in genetic algorithms. *en. Soft Comput.* **4**, 76–80 (July 2000).
8. Nachar, N. The Mann-Whitney U: A Test for Assessing Whether Two Independent Samples Come from the Same Distribution. *Tutorials in Quantitative Methods for Psychology* **4** (Mar. 2008).

References III

9. Osuna-Enciso, V., Cuevas, E. & Morales Castañeda, B. A diversity metric for population-based metaheuristic algorithms. *Information Sciences* **586**, 192–208. ISSN: 0020-0255. <https://www.sciencedirect.com/science/article/pii/S0020025521011956> (2022).
10. Radovanovic, M., Nanopoulos, A. & Ivanovic, M. Hubs in Space: Popular Nearest Neighbors in High-Dimensional Data. *Journal of Machine Learning Research* **11**, 2487–2531 (Sept. 2010).
11. Rastrigin, L. A. Systems of extremal control. *Nauka* (1974).
12. Rosenbrock, H. H. An Automatic Method for Finding the Greatest or Least Value of a Function. *The Computer Journal* **3**, 175–184. ISSN: 0010-4620. eprint: <https://academic.oup.com/comjnl/article-pdf/3/3/175/988633/030175.pdf>. <https://doi.org/10.1093/comjnl/3.3.175> (Jan. 1960).

References IV

13. Rudolph, G. *Globale Optimierung mit parallelen Evolutionsstrategien*. Diplomarbeit (Department of Computer Science, University of Dortmund, July 1990).
14. Srinivas, M. & Patnaik, L. Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics* **24**, 656–667 (1994).
15. Stark, N., Minetti, G. F. & Salto, C. *A new strategy for adapting the mutation probability in genetic algorithms*. in (2012).
<https://api.semanticscholar.org/CorpusID:55550829>.

References V

16. WELCH, B. L. THE GENERALIZATION OF "STUDENT'S"™ PROBLEM WHEN SEVERAL DIFFERENT POPULATION VARLANCES ARE INVOLVED. *Biometrika* **34**, 28–35. ISSN: 0006-3444. eprint: <https://academic.oup.com/biomet/article-pdf/34/1-2/28/553093/34-1-2-28.pdf>. <https://doi.org/10.1093/biomet/34.1-2.28> (Jan. 1947).

The End