

Adaptive Radiation as an Autotuning Strategy for Genetic Algorithms on Dynamic Problems

Heictor Costa

Bio-Inspired Computing Laboratory
Federal University of Pará
 Belém, Pará, Brazil
<https://orcid.org/0000-0003-3611-3675>

Roberto Oliveira

Bio-Inspired Computing Laboratory
Federal University of Pará
 Belém, Pará, Brazil
<https://orcid.org/0000-0002-6640-3182>

Abstract— As optimization processes have become more complex for embracing problems of different characteristics, it is necessary to have multiple adequate algorithms to tackle each of such problems. Bio-inspired evolutionary algorithms are suitable solutions for these situations, but they require re-tuning when working with different systems. Autotuning is a popular strategy to increase the adaptability of optimization algorithms. Adaptive Radiation (AR) is a phenomenon in nature that optimizes a population by diversity increase and niche specialization through intense mutation. This research aimed to insert this effect into the Genetic Algorithm (GA) workflow as a biological-inspired autotuning method, creating a new model called Genetic Algorithm with Adaptive Radiation (GAAR). The implementation of the AR component resulted in consistent and improved results on multiple benchmark functions from the CEC2019 challenge. The GAAR only changes the value of the AR component, which is enough to make this model achieve the best results in 57% of the tests and the worst results in 0% of the tests, while the Adaptive Particle Swarm Optimization (APSO) presented 39% and 12% of the best and worst results, respectively.

Keywords—Adaptive Radiation, Autotuning, Dynamic Problems.

I. INTRODUCTION

Modern computational systems, focused on optimizing complex problems, use bioinspired techniques for their superior capacity on working with nonlinear-multidimensional mathematical functions [1]. More specifically in this study field, a subgroup of algorithms implements the behavior observed in evolutionary phenomena of nature to optimize several problems, searching for an acceptable solution [2]. These are evolutionary algorithms.

The evolutionary algorithms work around the concept of a population, which progresses over generation steps as its timespan, generating better individuals. Such individuals are, then, interpreted as the solutions for the problem to be optimized [3]. In minimization processes, for example, the smaller valued individual is the best solution available.

The founder of the evolutionary algorithms is the Genetic Algorithm (GA), which mimics the basic steps of Darwinist Evolutionism [4]. In this model, the population selects its better individuals to crossover, and thereafter, introduces a mutation factor to simulate the stochastic behavior present in nature's observation.

Though revolutionizing the bioinspired field, the classic GA implementation became obsolete, generating the necessity of specific GA modalities, better adapted for each different problem. The elitism, for example, sets a percentage of the population, to not suffer mutation episodes, enabling the best individuals to be in the next generations, and increasing the

likelihood of finding the optimal solution, as it avoids random destruction of the best chromosomes and avoids premature convergence [5].

Although presenting improvement in the GA's results, this kind of process addition turns this algorithm into a high complexity, and large tunable parameter structure, creating opportunities for the introduction of more efficient, bioinspired optimizing algorithms. The Particle Swarm Optimization (PSO) is based on the birds' behavior in search for food, and it works with few parameters [6]. Differential evolution generates new descendants by perturbing the solutions with scaled difference vectors, it also presents a small number of tunable components, being a good strategy to find an acceptable solution to a problem of source inversion in practical environmental management [7].

Setting up an algorithm for each different problem, although a reasonable strategy, requires effort for try-and-error tuning of the parameters. This process becomes increasingly complex when applying certain algorithm to work over multiple problems of different qualities, in which the fitness functions are of different nature.

The bioinspired algorithms present the ability to adapt to dynamic situations, being the dynamism factor characterized by changing the mathematical function being optimized, therefore, replacing the problem itself. This is possible through the process of autotuning, which is the process of automatically adjusting algorithm parameters based on problem characteristics and performance metrics [8]. Autotuning allows bioinspired optimized algorithms to achieve optimal performance across a wide range of problem instances, allowing an algorithm, to be a powerful tool for multiple real-world applications.

Recalling the GA reference on evolutionary process, there are multiple biological actions related to genetic operations yet to be implemented in computational environment, such can ground autotuning behavior in the algorithm, as the adaptative inertia in the PSO [9]. However, multiple sub-processes running under the same algorithm requires higher computational effort. Thus, any bioinspired innovation demands a reasonable tradeoff, bringing more positive than negative traits to the algorithm.

This study looked for elaborating an autotuning method, that needs to change the value of a single parameter, but also relying on a coherent biological element, directly related to the behavior that inspired such algorithm.

References [10] and [11] highlight the presence of biodiversity and niche specialization in the survival journey, factors generated through episodes of intense genetic adaptation, later observed as being the starting point of

multiple species diversification. Both episodes are promoted by a phenomenon called Adaptive Radiation (AR), which provokes high mutation rates in endemic populations, allowing its individuals to quickly undergo adaptations in their characteristics for survival purposes.

Dealing with the necessity of optimizing multiple functions, aiming for high efficiency while also avoiding the necessity for re-tuning the operational parameters, this paper proposes the introduction of the AR into GA's execution, to allow this algorithm higher adaptation, with automatic tuning process, to different fitness function inputs. Such proposal will both improve GA's performance on the optimization results and present a better overall score in functions with different characteristics, due to the evolutive grounded autotuning method.

The remaining sections of this paper are presented as follows: Section II presents the Background, with state-of-art papers related to the study field of this research. Section III discusses the adopted Study Case, as the benchmark functions used in the performance tests. Section IV presents the concept of Adaptive Radiation phenomenon, and its effects on multiple species evolution. Section V explain each step of the proposed algorithm, detailing the method that led to each function's strategy, and the biological influence on the modeling of the AR function. Section VI presents the results from the proposed algorithm, and from its rival algorithms, and the performance explanation over the benchmark functions. Lastly, Section VII presents the conclusions of this study, citing the innovations carried out by this research, and the proposals for future works.

II. RELATED WORKS

Nowadays, when you look at artificial intelligence (AI) algorithms being developed, you get an interesting picture of how these algorithms work. There is a lot of manual work involved in automating the decision processes for the algorithms to perform well. It is AI tuning [12]. That is hard work, and it explains why an AI algorithm works great to solve a task, but if the task is switched, the performance will not be the same. In other words, fine-tuned AI algorithms are good for some tasks, but other tasks will require further tuning [13]. AI tuning is a promising field of study.

For evolutionary computing/swarm intelligence techniques, autotuning is also important [14]. Normally, evolutionary computation/swarm intelligence is used to tune some computational technique that is solving a problem [15][16]. But the performance of evolutionary computation/swarm intelligence algorithms is also dependent on a good tuning of their hyperparameters [17][18].

Among the evolutionary algorithms, the Genetic Algorithm (GA) is one of the oldest and most complex [19]. A characteristic of this complexity is the number of parameters [20][21]. The most important parameters of GA are population size, crossover rate, and mutation rate, among others. The mutation rate is responsible for exploring the search space and trying to maintain the diversity of the population [22][23]. Several versions of GA vary the mutation rate [24][25][26] or keep it high [27][28], to improve the performance of the algorithm.

More recently, the self-tuning strategy has been used in GA. In [29], the authors model the performance of machine learning algorithms as a function of the inherent structure of

very large datasets. Therefore, a parameterization mechanism is presented for an evolutionary rule-based machine learning system that can find the appropriate parameter value for a wide variety of synthetic classification problems with binary attributes and with/without added noise.

In [30], a comparison is made between four different tuning methods for GA hyperparameters. Non-automated method or grid search is used. The other three automated methods are iterated racing, mixed-integer parallel-efficient global optimization (MIP-EGO), and mixed-integer evolutionary strategies. The results suggest that even when interested in expected running time (ERT) performance, it might be preferable to use anytime performance measures for the configuration task. While minimizing ERT favors the average first hitting time of a single fixed target, maximizing the area under the empirical cumulative distribution function curve of running times (AUC) metric aims at optimizing anytime performance, which is measured across a whole set of (budget, target value) pairs. It shows that in several cases tuning for AUC yields configurations that have smaller ERT values than those that were obtained when directly tuning for ERT.

In [31] is proposed an autotuning genetic algorithm with two-dimension chromosomes for designing an optimal convolution neural network (CNN) model efficiently, to defect detection based on the surface images in manufacturing. A two-dimension chromosome is developed to represent CNN's structure and parameters. To enhance the search process, the crossover rate and mutation rate are tuned automatically according to the generation.

Given the characteristics of the AR phenomenon, and its coherent and useful implementation on optimization algorithms, this research adopted it as the main reference for an autotuning method. Since the AR is directly related to the evolutionary steps in nature, by adding a new mutational behavior to the population, the GA presents itself as the most capable algorithm to adopt this new component, for the evolutive abstraction present in this tool, creating a new model called Genetic Algorithm with Adaptive Radiation, or simply, GAAR, which balances the biodiversity and the niche specialization according to necessity.

III. STUDY CASE

To analyze the performance of the proposed algorithm, there were adopted as reference, trustable benchmark functions from the CEC-2019 Single Objective Optimization. Such functions were inherited from the 100-Digit Challenge [32], and it present 10 objective functions for optimization benchmarks. Every function was selected for this study, and they are presented in Table 1.

TABLE I. CEC-2019 FUNCTIONS

No.	Functions	$F_i = F_i(x^*)$	D	Search Range
1	Storn's Chebyshev Polynomial Fitting Problem	1	9	[-8192, 8192]
2	Inverse Hilbert Matrix Problem	1	16	[-16384, 16384]
3	Lennard-Jones Minimum Energy Cluster	1	18	[-4, 4]
4	Rastrigin's Function	1	10	[-100, 100]
5	Griewangk's Function	1	10	[-100, 100]
6	Weierstrass Function	1	10	[-100, 100]

7	Modified Schwefel's Function	1	10	[-100, 100]
8	Expanded Schaffer's F6 Function	1	10	[-100, 100]
9	Happy Cat Function	1	10	[-100, 100]
10	Ackley Function	1	10	[-100, 100]

The CEC functions present different mathematical characteristics in their own models, which carries out distinct scenarios for the optimization process. As presented in Figure 1, each function has its Number of Dimensions (D), Search Range, and the value of its respective global minimum. These functions simulate many complex environments for the proposed algorithm to work with, as a strategy to measure its performance, and evaluate at which opportunities it becomes the best optimization tool.

IV. ADAPTIVE RADIATION

Endemism is the term used in biology for a species or taxonomic group that is restricted to a particular geographic region [33]. Scientists find endemic species interesting because they offer valuable insights into the history and ecology of a region. For instance, many plant and animal species in Hawaii are endemic and not found anywhere else on the planet [34][35]. This suggests that the islands have been isolated for a long time, which has allowed unique life forms to evolve.

Several factors can contribute to the development of endemic species, including geographic isolation, climate, and competition with other species. Endemic species can also arise through AR, a process in which a single species diversifies into a variety of forms, each adapted to a specific ecological niche.

AR is a biological process that happens when a species diversifies into many descendant species, each adapted to a specific niche [36]. This usually happens when there is a major environmental change, such as the emergence of a new habitat or the extinction of other species. AR drives biodiversity by allowing organisms to exploit new opportunities and occupy new niches in the ecosystem efficiently, thanks to a mutation surge [37].

As the AR is a product of endemism, the radiation impact may be measured by similar metrics to the endemic units. Reference [38] presented a study of different metric relations, which indicates the most effective for each scenario, proving that for Discrete Study Units and Phylogenic Data, which is the case for the proposed methodology on this current study, the Phylogenic Diversity-Endemism (PDE) is the most effective metric. This relation is of immense contribution to this research, being the main reference for the mathematical model of the AR function presented in the Methodology section.

V. METHODOLOGY

On this section, it is presented the methodology of the algorithm development process, it focuses on presenting and describing the steps of the proposed algorithm as to make sense of its workflow.

Most of the steps presented in this section are similar to classic GA implementations from other papers, since the goal of this research is to bring a minimal effort implementation contribution, which draws most of the innovation to the Adaptive Radiation step.

A. Population Initialization

The function responsible for generating the initial population demands 2 inputs, the number of individuals (I_n) and the number of dimensions for every individual (D). When started, this step generates I_n random solutions, each made of D values, uniformly distributed in between the Search Range boundaries.

This paper presents a playful illustration strategy for observing population progress based on the Agarose Gel Electrophoresis method with Ethidium bromide to separate and observe genetic patterns. See Figure 1.

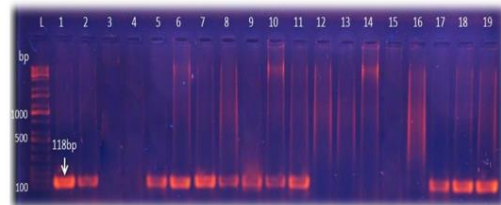


Fig. 1. Agarose Gel Electrophoresis. [39]

Figure 1 shows a gene present in multiple samples, helping identify certain proteins in different individuals. The paper uses a similar method to show the population generated by the algorithm, as seen in Figure 2.

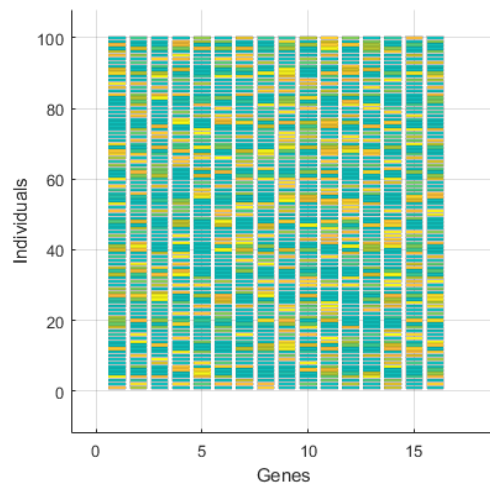


Fig. 2. Population Genetic Pattern Visualization - Upper View.

Figure 2 shows as example a population with 100 individuals and 16 Genes. The colors of this illustration are relative to the values of each gene, which are better represented in an isometric view, as shown in Figure 3.

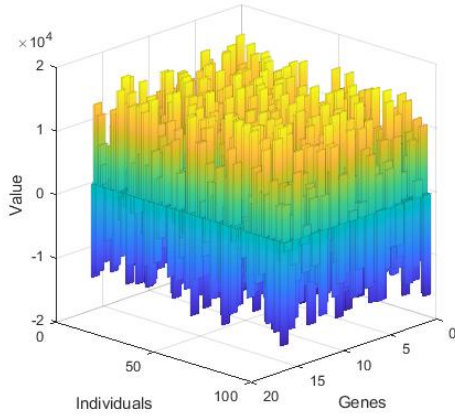


Fig. 3. Population Genetic Pattern Visualization - Isometric View.

It is important to notice that Figures 2 and 3 represent the same population, only switching between viewing angles. The Population Initialization function generates the data necessary for producing the genetic pattern plot presented in the Results section.

B. Fitness Evaluation

To measure the quality of an individual, thus, how good is a solution, it is measured its response to a mathematical function, known as Fitness Function. In this paper, the Fitness Functions adopted are the 10 CEC2019 models presented in the study case section. The Fitness Evaluation process is the first step of each generation in the loop.

This paper's approach to score the quality of a solution is based on its proximity to the global solution, being the measurement the direct result of an individual to the function. It is known the global solution for each function adopted is $f = 1$ [32].

C. Choice process of the Selection Function

A recent comparative study between the Proportionate Roulette Wheel, Linear and Exponential Ranking, and Tournament selection methods in a GA presents the Tournament method as the most efficient, for presenting less time complexity ($O(n)$), and no need for a sorting method [40]. This method also presents more biologically balanced behavior, for reducing the possibility of takeover by dominant individuals, which reduces premature convergence odds. Thus, the Tournament Selection was adopted as the selection method for this research.

This method performs multiple pairwise comparatives on the population, selecting only the individual with best fitness value of each pair, which reduces the population size in 50% after this step [41].

D. Strategy Comparative Organization

This research tested the mutation methods Permutation (P), Scramble (S), and Reset (R), and the crossover methods Single-Point Cut (SP), Double-Point Cut (DP) and Uniform-Point Cut (UP). Each strategy combination is named after the concatenation of Mutation and Crossover abbreviations, e.g., Permutation with Uniform-Point Cut it's recognized by the abbreviation PUP, being all the strategies shown in Table 2.

TABLE II. GA STRATEGIES

	Permutation (P)	Scramble (S)	Reset (R)
Single-Point Cut (SP)	PSP	SSP	RSP
Double-Point Cut (DP)	PDP	SDP	RDP
Uniform Point Cut (UP)	PUP	SUP	RUP

E. Crossover Function Method Choice

For this algorithm development, it was adopted to work with constant population, for that, the crossover function must restore the number of individuals by having each couple to produce 4 children. The preference for maintaining a constant population relies on biological references, since a species efficiency is often related to its capability of density regulation [42][43]. Also, there is a linear relation between population size and fitness capacity, so if a population reduces immensely, it loses its protection against environmental permutation, increasing the rates of extinction [44].

SP method involves selecting a single cut point in both parents' chromosomes and swapping the segments after the cut point. While the DP crossover involves selecting two cut points in both parents' chromosomes and swapping the segments between the two cut points. At last, the UP crossover approach involves selecting cut points at random along both parents' chromosomes and swapping the segments between the cut points [45].

The efficiency of each cut point method relies on the specific problem being solved and the characteristics of the search space. The effectiveness of each method can also vary depending on the specific genetic operators used and the parameters chosen for the genetic algorithm, thus the strategies comparatives come out as a manner to base the crossover method choice.

F. Mutation Function Method Choice

Nature's noisy stochastic factor is embedded in GA's steps by the introduction of Mutation Function, this component affects a percentage of the population, multiplying the genes of such by random coefficients. This process increases the similarity of the computational algorithm to its biological reference, as to approach the endless variables which interfere in the ecological environment.

As detailed in the previous sub-section, this research compared the performance of 3 different strategies (P, S, and R). The Permutation mutation method involves selecting two positions in the chromosome, at random, and swapping the elements at those positions [46]. Scramble mutation method involves selecting two positions in the chromosome at random and scrambling the elements between those positions [47]. At last, the Reset mutation method involves selecting a position in the chromosome at random and resetting the element at that position to a specified value [48], however for this research, it was adopted to work with 2 random positions in the R method, and the new value is randomly generated inside the search space.

For every mutation method, the number of affected individuals is set by the Mutation Rate (M) variable value. The value was set as constant for all the benchmark tests.

G. Adaptive Radiation Function Formulation

As shown in Section IV, the AR diversifies the population while adapting individuals to new environmental

configurations niches, action promoted by an intense mutation burst. Given the concepts introduced about the AR phenomenon, its implementation model aimed to cause to the algorithm's population, similar symptoms to the ones perceived in natural behaviors.

To avoid dubieties on the impact of the AR Function, it is important to compare its operational differences to the classic mutation function. The major dissemblance between those is their application strategy, due to their biological inspiration, as further explained.

Based on the workflow of the Mutation Function, detailed on the previous section, at each iteration, for each affected individual, 2 random numbers are generated, replacing 2 previous genetic variables. This random number generation procedure occurs individually, thus, the random seed for each individual has time to progress avoiding repetition or high similarity. The natural mutation process happens mostly individually, with unique settings and results. Thus, such computational characteristic is fundamental in this stage as to simulate every day biological elements combinations which, directly or indirectly, interfere in the genetic changes.

The AR procedure, otherwise, does not repeat on an individual's basis, it happens once at each generation, affecting a variable number of individuals over the optimization process. The variable responsible for indicating how much of the population will be affected by the AR is called Dynamic Rate (DR), and its value is automatically adjusted at each generation by the algorithm, as shown further in this section.

The AR generates a DR-by-D matrix of random values, which amplitude boundaries are $[-1,1]$, the random valued matrix multiplies the matrix of affected individuals, promoting collective changes from the same seed. The amplitude values are set for statistically equate to minimum and maximum concordance, respectively, which represents the acceptance (stochastic correlation) of a current gene into this environmental shift introduced with the AR effect.

As presented in the Adaptive Radiation section, the PDE is the most compatible metric for this research, as it measures the proportion of phylogenetic diversity restricted to a study region [38][49]. The PDE measurement is presented in Equation 1.

$$PDE = \frac{\sum_{e=1}^E L_e}{\sum_{s=1}^S L_s} \quad (1)$$

Being E the endemic richness, S the total species richness, L_s the phylogenetic branch lengths of all the resident species, and L_e the phylogenetic branch lengths of the endemic species.

The DR is a value responsible for indicating the number of individuals affected by the AR, its value changes based on the necessity for the population to evolve, comparing the current generation to the initial one, thus, their ancestor. The DR is given by the value of a factor δ , mathematically modeled after the PDE, which consists of the percentual relation between the variance of the current and first populations, as presented in Equation 2 and 3.

$$\delta = \frac{\sum_{i=1}^p \sigma^2(Pp)}{\sum_{i=1}^p \sigma^2(Pp_0)} \quad (2)$$

$$DR = \begin{cases} \delta, & \text{if } \delta \leq 100\% \\ 100\%, & \text{otherwise} \end{cases} \quad (3)$$

Where Pp is the current population, and Pp_0 the initial population, and σ^2 is the variance defined by the Equation 4.

$$\sigma^2 = \frac{\sum_{i=1}^p (x_i - \mu)^2}{ps} \quad (4)$$

Being x_i the i^{th} individual from the population, μ the population mean, and ps is the population size.

This approach provides an excellent way to compare the genetic dispersion of populations while simulating biological phenomena. The initial generation is uniformly distributed throughout the search space, causing the DR to reach its peak value in the initial stages of the algorithm, as the population has no evolutive niche progression yet. This increases the mutation effects in the population, mimicking the effects of an environmental alteration.

After the AR affects the current population, the generation ends, and the new generation begins, starting the next iteration in the loop with the Fitness Evaluation step.

The GAAR follows much of the regular GA workflow, with the addition of the AR function. The GAAR steps are shown in Figure 4.

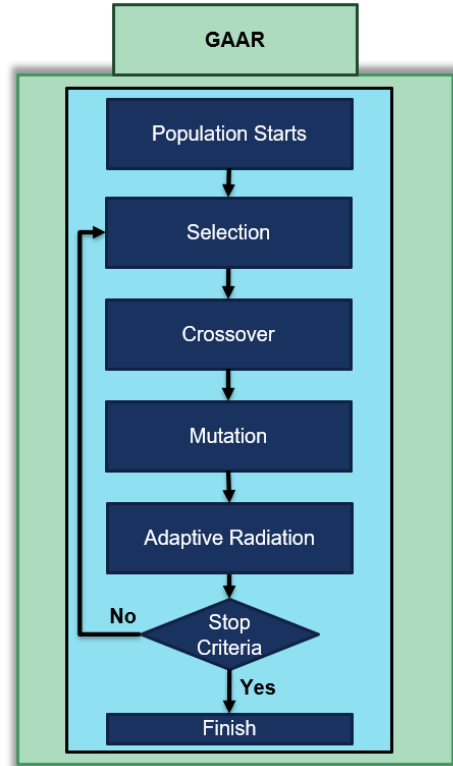


Fig. 4. GAAR Workflow.

H. Algorithm Comparative

An important part of this research methodology is comparing the proposed algorithm to some rival approaches, besides using the GAAR over the CEC2019 functions, this study adopted the as competitors the Classic GA (GA), Classic Particle Swarm Optimization (PSO), and Adaptive Particle Swarm Optimization (APSO).

The GA was chosen as a direct reference to the GAAR, adopting similar parameters (Table 3), being the AR usage their only difference. The PSO was also selected as a rival for this methodology for, although being a populational bioinspired algorithm, presenting essentially different characteristics, as being non-evolutive, cooperative, and considerably more exploratory. At last, the APSO was included as an equivalent AR component to the PSO, for adaptative behaviors comparative, however, adopting the same values to the other parameters (Table 4).

TABLE III. GA AND GAAR PARAMETERS

Parameter	GA	GAAR
Population Size	1000	1000
Mutation Rate	5%	5%
Generations	300	300
Crossover Method	Double-Point Cut	Double-Point Cut
Mutation Method	Reset	Reset
AR Type	Null	Dynamic

TABLE IV. PSO AND APSO PARAMETERS

Parameter	PSO	APSO
Swarm Size	1000	1000
Inertia Weight	1	[0.1,1]
Self-Adjustment Weight	1.60	1.60
Social Adjustment Weight	1.20	1.20
Iterations	300	300

After setting up the benchmark environment, this research performed multiple tests on the fitness functions, and it was perceived that 20 tests were an optimal value to obtain reliable data, since it was enough to comprise the whole magnitude of different performances of the algorithms, caused by stochastic motivators. The results and observations from these tests are detailed in the next section.

VI. RESULTS

The first result of the proposed research is a product from the Strategy Comparative step. This result was useful for selecting the best alternative, between the 9 strategies available. The results of a singular execution of the functions are presented in Figure 5, in the format of a Spider Plot [50].

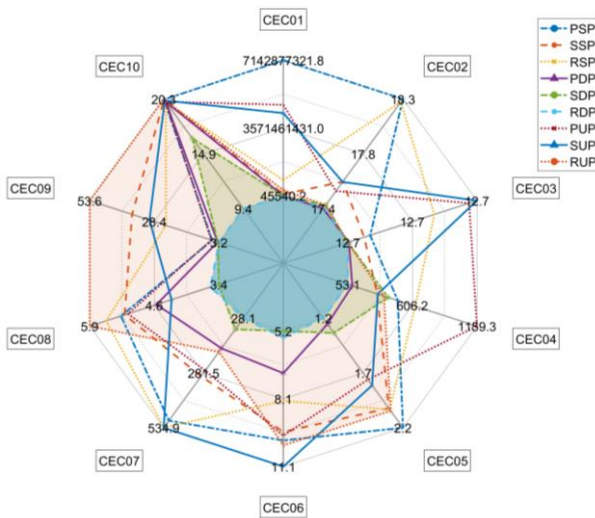


Fig. 5. Strategy Results on the CEC2019 Functions.

The values closer to the center are the best (minimum values) results for each fitness function, and the more distant are the worst results, thus the best strategy is the one to present most of its results closer to 1. Following this analysis method, the RDP was the strategy that most times achieved the best solutions. This behavior is an overall illustration of the whole test group, for representing most of the results pattern.

For the crossover method, the total results present the DP achieving the best results in 92% of the tests, thus, being elected as the best strategy for this research. The benchmark data also informs the Reset was the best mutation method, achieving the best results in 44% of the tests, while the Scramble and Permutation methods appear with 28% and 25% efficiency rates, respectively, and the last 3% is distributed over irregular disperse alternatives.

The quantity of executions to compound the results was necessary to avoid temporary divergency or measures based on a singular episode of high error, mostly caused by a bad seed. Figure 6 samples the results of comparatives between the adopted algorithms, with a spider plot approach [50], thus, the smaller values are closest to the center, while the higher values are on the periphery of the web grid.

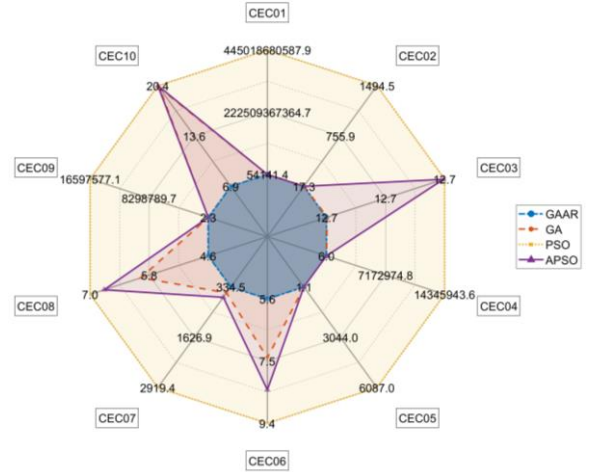


Fig. 6. Algorithms Results on the CEC2019 Functions.

As exemplified by Figure 6, the GAAR presents itself as being the most efficient algorithm, followed by the APSO. The whole data collected by the benchmarks, shows the GAAR providing the best solution in 57% of the tests, the APSO in 39%, and GA in 4%. It is important to highlight that the GAAR never presented the worst performance, for any function, while the PSO presented 88% of the worst solutions, and the APSO 12%.

The data from the tests performed, showed that the GAAR mostly loses to the APSO in functions 02, 04, 05 and 09, with few exceptional episodes ($\cong 3\%$). Such functions presented itself as being a more comfortable scenario for the APSO, given its major exploratory behavior. For better understanding the results, this paper explains the algorithms' relations to the functions' most influential characteristics.

CEC01: The most complex function in its group, it has the highest level of conditionality, modeled after three independent equation systems based on signal pulses covering nine dimensions. Each system presents different intervals and behaviors, drastically altering the function's solution space when traversing a continuous, unidirectional path over its

surface. It requires greater precision tuning as the population approaches the global solution. PSO performed the worst among the algorithms, being the most exploratory in nature, while GAAR achieved the best performance due to its self-adaptive nature (Dynamic Radiation). A consequence of such conditionality is observed on the shape of the population at the last generation, as exhibited on Figure 7.

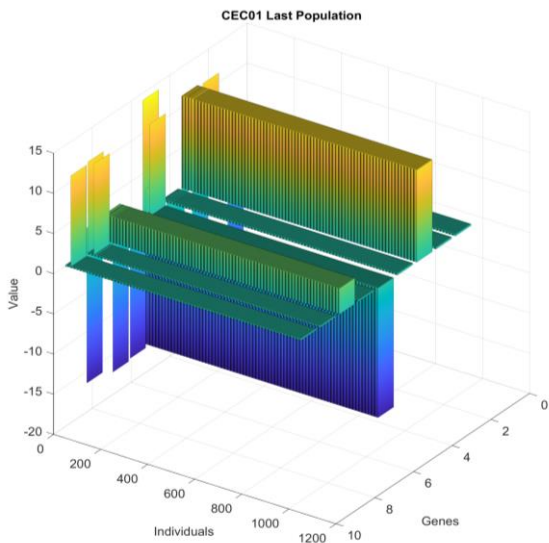


Fig. 7. CEC01 Last Generation Genetic Pattern Visualization - Isometric View.

Figure 7 visually demonstrates a noticeable pattern in the genetic adaptation of the population, as it reaches optimal values. It is apparent that, even without intentional weighting of genetic preferences, the algorithm focuses on minimizing certain genes with a priority scheme. However, the amplitude values of the presented genes remain far from the search space boundaries.

CEC02: The APSO produced the best results by optimizing the Hilbert Inverse Matrix. This is because the algebraic relation between three principal components (two functions and one matrix) creates a standardized nature that benefits algorithms evaluating the best value found by one individual or even a small group within the population.

CEC03: No large difference presented between the algorithm's results. The function being optimized is not complex, as the energy values of atomic interactions change based on the distance between the particles. The optimization problem involves 6 atoms, but 4 of them have trivial solutions, meaning that the optimal parameter values are not unique. Unlike CEC01, the search orientation does not affect the optimization's efficiency. The function's complexity arises from the accidental overlapping of atoms, which causes an infinite energy value. However, this value is not computed, leading to a reduction in the total solution value in an inaccurate way.

CEC04: The Rastrigin function was a challenge for GAAR because of the distance between the penultimate and global optima, providing most efficiency of more adaptative-exploratory algorithms, fact observed by the results achieved by the APSO, unreachable by their opponents. The generating function is simple, but the topology includes the sum of D cosine functions (where D is the number of dimensions), resulting in a dense function with pseudo-prosperous regions.

CEC 05 and 09: These functions have mostly smooth surfaces but are flatter near the optimum point. Algorithms with a component dedicated to the individual result could achieve this characteristic more efficiently. APSO algorithm obtained the best result in both functions. However, the GAAR strategy remains a good choice for optimizing these functions.

CEC 06, 07, and 08: High oscillation is observed in certain amplitude ranges, leading to many local optima within the search interval. GAAR consistently performed the best across all these functions, with particularly strong results for CEC07, which presents systems of equations. On the other hand, PSO and APSO struggled with function 08 due to its prosperity region being limited to a small area of the search space. However, it did exhibit high topological density and low oscillatory amplitude with respect to the value of $f(x)$.

CEC10: GAAR was the best algorithm in all executions, despite the function's local minima and pseudo-prosperity regions. The intense effect of adaptive radiation in initial generations quickly pulls individuals out of local minima, promoting greater population agitation. Although not the fastest, individuals that enter the region of prosperity reach high levels of intensification, resulting in excellent optimal values.

VII. CONCLUSION

In this paper, we present a novel approach to genetic algorithms for dynamic problems by implementing an adaptive radiation step. Our approach is based on the biological phenomenon of adaptive radiation, which allows populations to diversify while adapting to new environmental niches. We also propose a mathematical model for the adaptive radiation function and compare its performance to classic genetic algorithms and other bio-inspired algorithms, such as particle swarm optimization.

As this study aims for the most adaptable algorithm, we present a strategy for a comparative step to select the best crossover and mutation methods, which proved to be effective in improving the GAAR's performance, based on these methods' results over the benchmark functions. Our results show that the proposed model, the GAAR, outperforms the other algorithms in most of the benchmark tests, presenting itself as the most consistent, accurate, and stable algorithm over the tests.

Future work includes testing the GAAR on other dynamic problems, such as optimal dispatch on power grids, focusing on hybrid fuel systems, which require dynamic changes in their operational parameters. We also plan to investigate strategies to increase the GAAR model's proximity to biological references, while avoiding computationally inefficient operations, to perpetuate this algorithm as a high-end tool.

Overall, we have presented that our approach has the potential to significantly improve the performance of genetic algorithms on problems of different nature and behavior, making them more effective tools for solving real-world optimization dynamic problems.

ACKNOWLEDGMENT

We are thankful to the Federal University of Pará (UFPA), to the Graduate Program in Electrical Engineering of the

Institute of Technology (PPGEE - ITEC), and to the Higher Education Personnel Improvement Coordination (CAPES).

REFERENCES

- [1] Darwish, A. "Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications". *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 231-246, December, 2018. DOI: 10.1016/j.fcij.2018.06.001
- [2] L. Cao, H. Chen, Y. Chen, Y. Yue, and X. Zhang, "Bio-Inspired Swarm Intelligence Optimization Algorithm-Aided Hybrid TDOA/AOA-Based Localization," *Biomimetics*, vol. 8, no. 2, p. 186, Apr. 2023, doi: 10.3390/biomimetics8020186.
- [3] J. Li and L. Li, "An Improvement Proposal of Genetic Algorithms Based on Information Entropy and Game Theory," 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), Granada, Spain, 2019, pp. 36-43, doi: 10.1109/SNAMS.2019.8931847.
- [4] J. H. Holland. *Adaptation in Natural and Artificial Systems*, 2nd ed. Ann Arbor: MIT Press, 1992.
- [5] B.V., Natesha, R. M. R., Guddeti. "Adopting elitism-based Genetic Algorithm for minimizing multi-objective problems of IoT service placement in fog computing environment". *Journal of Network and Computer Applications*, vol. 178, pp. 102972, March, 2021. DOI: 10.1016/j.jnca.2020.102972.
- [6] R., Cortez, R., Garrido R., Mezura-Montes. "Spectral Richness PSO algorithm for parameter identification of dynamical systems under non-ideal excitation conditions". *Applied Soft Computing*, vol. 128, no. 1, pp. 109490, October, 2022. DOI: 10.1016/j.asoc.2022.109490.
- [7] S., Mao, F., Hu, J., Lang, T., Chen, S., Cheng. "Comparative Study of Impacts of Typical Bio-Inspired Optimization Algorithms on Source Inversion Performance". *Frontiers in Environmental Science*, vol. 10, no.1, pp. 1-14, June, 2022. DOI: 10.3389/fenvs.2022.894255.
- [8] PapRajani, N., Kumar, K. S., Kaswan. "Soft Hunting Algorithm for Auto-Tuning Software Reliability Growth Models". *Wireless Communications and Mobile Computing*, vol. 3752264, no. 1, pp. 1-10, August, 2022. DOI: 10.1155/2022/3752264
- [9] G. M. Nayeem, M. Fan and Y. Akhter, "A Time-Varying Adaptive Inertia Weight based Modified PSO Algorithm for UAV Path Planning," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), DHAKA, Bangladesh, 2021, pp. 573-576, DOI: 10.1109/ICREST51555.2021.9331101.
- [10] A., Rapini, C., Bitencourt, F., Luebert, D., Cardoso. An escape-to-radiate model for explaining the high plant diversity and endemism in campos rupestres. *Biological Journal of the Linnean Society*, vol. 133, no. 2, pp. 481-498, September, 2020, DOI: 10.1093/biolinnean/blaa179
- [11] E. J., Richards, J. A., McGirr, J. R., Wang, M. E., St John, J. W., Poelstra, M. J., Solano, D. C., O'Connell, B. J., Turner, C. H., Martin. "A vertebrate adaptive radiation is assembled from an ancient and disjunct spatiotemporal landscape". *Proceedings of the National Academy of Sciences of the United States of America*, vol. 118, no. 20, pp. e2011811118. May, 2021, DOI: 10.1073/pnas.2011811118
- [12] Victoria, A. H.; Maragatham, G. Automatic tuning of hyperparameters using Bayesian optimization. *Evolving Systems* 12, 217-223 (2021). <https://doi.org/10.1007/s12530-020-09345-2>
- [13] R. R. Karn; P. Kudva; I. A. M. Elfadel. Dynamic Autoselection and Autotuning of Machine Learning Models for Cloud Network Analytics. *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 5, pp. 1052-1064, 1 May 2019, doi: 10.1109/TPDS.2018.2876844.
- [14] Cheng, M.-Y.; Fang, Y.-C.; Wang, C.-Y. Auto-tuning SOS Algorithm for Two-Dimensional Orthogonal Cutting Optimization. *KSCE J Civ Eng* 25, 3605-3619, 2021. <https://doi.org/10.1007/s12205-021-0522-y>
- [15] U. R. Acharya; M. R. K. Mookiah; S. Vinitha Sree; R. Yanti1, R. J. Martis; L. Saba; F. Molinari; S. Guerriero; J. S. Suri. Evolutionary Algorithm-Based Classifier Parameter Tuning for Automatic Ovarian Cancer Tissue Characterization and Classification. *Ultraschall in Med*, 35: 237-245, 2014. <http://dx.doi.org/10.1055/s-0032-1330336>
- [16] Marcelo Favoretto Castoldi; Sérgio Carlos Mazucato Júnior; Danilo Sipoli Sanches; Carolína Ribeiro Rodrigues; Rodrigo Andrade Ramos. Automatic Tuning of PSSs and PODs Using a Parallel Differential Evolution Algorithm. *International Journal of Natural Computing Research (IJNCR)*, 4(1), 2014. DOI: 10.4018/ijnrcr.2014010101
- [17] Lin, L., Gen, M. Auto-tuning strategy for evolutionary algorithms: balancing between exploration and exploitation. *Soft Comput* 13, 157-168 (2009). <https://doi.org/10.1007/s00500-008-0303-2>
- [18] Roberto Ugolotti; Stefano Cagnoni. Automatic Tuning of Standard PSO Versions. *Conference on Genetic and Evolutionary Computation - GECCO'15*, July, pp 1501-1502, 2015. <https://doi.org/10.1145/2739482.2764706>
- [19] Lance D. Chambers; Lance D. Chambers. *Practical Handbook of Genetic Algorithms - Complex Coding Systems - Volume III*. CRC Press, 592 pages, 2019.
- [20] P. Pongcharoen; C. Hicks; P. M. Braiden.; D. J. Stewardson. Determining optimum Genetic Algorithm parameters for scheduling the manufacturing and assembly of complex products. *International Journal of Production Economics*, Volume 78, Issue 3, Pages 311-322, 2002. [https://doi.org/10.1016/S0925-5273\(02\)00104-4](https://doi.org/10.1016/S0925-5273(02)00104-4)
- [21] A. Vlasov; A. Khomchenko; A. Faizliev; S. Mironov; A. Grigoriev. Parameter tuning of a genetic algorithm for finding central vertices in graphs. *Journal of Physics: Conference Series*, 1784, 012009, 2021. doi:10.1088/1742-6596/1784/1/012009
- [22] Watson, T.; Messer, P. Increasing Diversity in Genetic Algorithms. *Developments in Soft Computing - Advances in Soft Computing*, John, R., Birkenhead, R. (eds), vol 9, Physica, Heidelberg, 2001. https://doi.org/10.1007/978-3-7908-1829-1_14
- [23] McGinley, B.; J. Maher; C. O'Riordan; F. Morgan. Maintaining Healthy Population Diversity Using Adaptive Crossover, Mutation, and Selection. *IEEE Transactions on Evolutionary Computation*, 15:692-714, 2011. DOI: 10.1109/TEVC.2010.2046173
- [24] Hassanat, A.; Almohammadi, K.; Alkafaween, E.; Abunawas, E.; Hammouri, A.; Prasath, V.B.S. Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach. *Information* 2019, 10, 390. <https://doi.org/10.3390/info10120390>
- [25] Darren M. Chitty. 2022. Accelerating genetic algorithm evolution via ant-based mutation and crossover for application to large-scale TSPs. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO '22)*. Association for Computing Machinery, New York, NY, USA, 2046-2053. <https://doi-org.ez3.periodicos.capes.gov.br/10.1145/3520304.3534020>
- [26] Giti Javadi; Ehsan Aminian. A New Method for Tuning Mutation and Crossover Rate in Genetic Algorithm. 9th International Conference on Machine Learning and Computing - ICMLC 2017, February, pp 217-220, 2017. <http://dx.doi.org/10.1145/3055635.3056622>
- [27] N.-E. Croitoru. High Probability Mutation and Error Thresholds in Genetic Algorithms. 17th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Timisoara, Romania, pp. 271-276, 2015. doi: 10.1109/SYNASC.2015.51.
- [28] Juraj Spalek; Michal Gregor. Adaptive Switching of Mutation Rate for Genetic Algorithms and Genetic Programming. *Applied Computer Science*, Volume 7, Number 1, 2011.
- [29] M. A. Franco; N. Krasnogor; J. Bacardit. Automatic Tuning of Rule-Based Evolutionary Machine Learning via Problem Structure Identification. *IEEE Computational Intelligence Magazine*, vol. 15, no. 3, pp. 28-46, Aug. 2020. doi: 10.1109/MCI.2020.2998232.
- [30] F. Ye; C. Doerr; H. Wang; T. Bäck. Automated Configuration of Genetic Algorithms by Tuning for Anytime Performance. *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 6, pp. 1526-1538, Dec. 2022. doi: 10.1109/TEVC.2022.3159087.
- [31] Huynh, N.-T.; Ho, D.-D.; Nguyen, H.-N. An Approach for Designing an Optimal CNN Model Based on Auto-Tuning GA with 2D Chromosome for Defect Detection and Classification. *Sustainability*, 15, 5455, 2023. <https://doi.org/10.3390/su15065455>
- [32] K. V. Price, N. H. Awad, M. Z. Ali, P. N. Suganthan, "Problem definitions and evaluation criteria for the 100-digit challenge special session and competition on single objective numerical optimization," Technical Report, Nanyang Technological University, Singapore, November 2018.
- [33] Staude, I. R., Navarro, L. M., Pereira, H. M. "Range size predicts the risk of local extinction from habitat loss". *Global Ecology and Biogeography*, 29(1), 16-25. September, 2020. DOI:10.1111/geb.13003
- [34] Gillespie, R. G., Croom, H. B., & Palumbi, S. R. Multiple origins of a spider radiation in Hawaii. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 91, no. 6, pp: 2290-2294. November, 1994. DOI: 10.1073/pnas.91.6.2290

- [35] Losos J. B. Adaptive radiation, ecological opportunity, and evolutionary determinism. American Society of Naturalists E. O. Wilson award address. *The American naturalist*, vol. 175, no. 6, pp: 623–639. June, 2010. DOI: 10.1086/652433
- [36] Richards, E. J., McGirr, J. A., Wang, J. R., St John, M. E., Poelstra, J. W., Solano, M. J., et al. (2021). A vertebrate adaptive radiation is assembled from an ancient and disjunct spatiotemporal landscape. *Proc. Natl. Acad. Sci. U.S.A.* 118 (20), e2011811118. doi:10.1073/pnas.2011811118
- [37] Berner, D., Salzburger, W. The genomics of organismal diversification illuminated by adaptive radiations. *Trends in Genetics*, vol. 31, no. 9, pp: 491-499. August, 2015. DOI: 10.1016/j.tig.2015.07.002.
- [38] Shipley, B. R., & McGuire, J. L. (2021). "Interpreting and integrating multiple endemism metrics to identify hotspots for conservation priorities". *Biological Conservation*, vol. 265, no. 1, pp: 109403. DOI: 10.1016/j.biocon.2021.109403.
- [39] Enas Jalil Al-Mayali.et. al. "Molecular Detection of Bio Film Gene in Pseudomonas Aeruginosa Isolates from Tonsillitis". *Journal of Global Pharma Technology*, vol. 10, no. 03, pp: 57-64. 2018.
- [40] A. Bala and A. K. Sharma, "A comparative study of modified crossover operators," 2015 Third International Conference on Image Information Processing (ICIIP), Wagnaghat, India, 2015, pp. 281-284, doi: 10.1109/ICIIP.2015.7414781.
- [41] A. Shukla, H. M. Pandey and D. Mehrotra, "Comparative review of selection techniques in genetic algorithm," 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), Greater Noida, India, 2015, pp. 515-519, doi: 10.1109/ABLAZE.2015.7154916.
- [42] Shao, J. W., Zhang, X. P., Zhang, Z. X., & Zhu, G. P. "Effects of population size on reproductive success of the endangered and endemic species *Primula merrilliana*". *Journal of integrative plant biology*, vol. 50, no. 9, pp. 1151–1160. 2008. <https://doi.org/10.1111/j.1744-7909.2008.00655.x>
- [43] Araya-Ajoy, Y. G., Niskanen, A. K., Froy, H., Ranke, P. S., Kvalnes, T., Rønning, B., Pepke, M. L., Jensen, H., Ringsby, T. H., Saether, B. E., & Wright, J. "Variation in generation time reveals density regulation as an important driver of pace-of-life in a bird metapopulation". *Ecology Letters*, 24, 2077– 2087, 2021 <https://doi.org/10.1111/ele.13835>
- [44] Reed, D. H. "Relationship between Population Size and Fitness". *Conservation Biology*, vol. 19, no. 2, pp. 563–568, 2005. <http://www.jstor.org/stable/3591269>
- [45] A. Bala and A. K. Sharma, "A comparative study of modified crossover operators," 2015 Third International Conference on Image Information Processing (ICIIP), Wagnaghat, India, 2015, pp. 281-284, doi: 10.1109/ICIIP.2015.7414781.
- [46] R. C. Purshouse and P. J. Fleming, "On the Evolutionary Optimization of Many Conflicting Objectives," in *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 770-784, Dec. 2007, doi: 10.1109/TEVC.2007.910138.
- [47] Soni, N., & Kumar, T. "Study of Various Mutation Operators in Genetic Algorithms". *International Journal of Computer Science and Information Technologies*, vol. 5, no. 3, 2014, 4519-4521
- [48] Srivatsa D, Teja TPVK, Prathyusha I, Jeyakumar G (2019) An empirical analysis of genetic algorithm with diferent mutation and crossover operators for solving Sudoku. In: B. Deka, P. Maji, S. Mitra, D. Bhattacharyya, P. Bora, and S. Pal (eds), *Proceedings of the 8th international conference on Pattern Recognition and Machine Intelligence (PREMI)*. Lecture Notes in Computer Science, vol 11941, Springer, Cham, pp 356–364, Tezpur, India
- [49] Faith, D.P., Reid, C. A. M., Hunter, J. "Integrating Phylogenetic Diversity, Complementarity, and Endemism for Conservation Assessment". *Conservation Biology*. vol. 18, no. 1, pp.255–261, 2004. DOI: <https://doi.org/10.1111/j.1523-1739.2004.00330.x>
- [50] Moses (2023). spider_plot (https://github.com/NewGuy012/spider_plot/releases/tag/20.2), GitHub.