A new breeding crossover approach for evolutionary algorithms

J. C. Felix-Saul, Mario García-Valdez

Abstract In a previous work, we introduced a population-based, bio-inspired algorithm. The proposed algorithm is inspired by the biological animal life cycle, consisting of the stages of birth, growth, reproduction, and death. Our algorithm was initially based on the canonical Genetic Algorithm (GA), where all the individuals have a genotype (chromosome). One difference to highlight in our algorithm is that both the crossing and the mutation are executed through independent processes that randomly affect the population. This paper focuses on breeding, whereas in earlier versions of the algorithm, we used the traditional GA one-point crossover. In this paper, we propose a different alternative to the classical approach, where part of the genetic information is directly copied to each of the offspring in the crossover operator, where this type of crossover may not perform well in continuous optimization problems. In this proposal, we use the parent's genetic information for each gene, using those values as lower and upper bounds of a range, where a random value within that range determines the new value for that gene index of the offspring. This is similar to what is used by algorithms such as Differential Evolution, where we consider our proposal as a variation of existing proposals. We expect this new operator to allow the offspring to continue exploring new search spaces with the birth of individuals. In this paper, we use the benchmark functions introduced in the Competition on Evolutionary Computation for the 2017 edition (CEC2017) to compare the traditional one-point crossover and our proposed strategy. Experimental results indicate that our proposed operator may be a good alternative for the canonical crossover.

J. C. Felix-Saul
TecNM, Tijuana Institute of Technology, Tijuana, Mexico, e-mail: jose.felix201@tectijuana.edu.mx

Mario García-Valdez
TecNM, Tijuana Institute of Technology, Tijuana, Mexico, e-mail: mario@tectijuana.edu.mx

1 Introduction

Biologically inspired algorithms are a very effective technique to solve complex optimization problems[1, 2, 3]. Traditionally, nature-inspired algorithms are developed with a sequential perspective[4, 5], meaning that all tasks execute one step at a time, where all processes must wait for the previous to finish before continuing (synchronous perspective). Some architectures address this issue by working on the cloud[6, 7, 8] and finding solutions on distributed technologies.

In previous work, we introduced a population-based, bio-inspired algorithm[9, 10]. The proposed algorithm is inspired by the biological animal life cycle, consisting of the stages[11] of birth, growth, reproduction, and death. Our algorithm was initially based on the canonical Genetic Algorithm (GA)[12, 13], where all the individuals have a genotype (chromosome). Our algorithm's goal is to mimic the animal life cycle, where at any given moment, new individuals are born to be part of the population and participate in the collective evolution. As time passes, they grow older and mature, suffering changes throughout their lives that we chose to represent as mutations.

In our analysis, we thought of death's work to maintain balance in the population by enforcing the survival of the fittest. As in life, death can happen to everyone: from a newborn to the elderly, where fitness will determine the individual's longevity. This algorithm was inspired by the traditional Genetic Algorithm [12, 13], meaning that all individuals have a genotype (chromosome) that is a list of values. We calculate the individual's fitness with the evaluation function and do crossover and mutation to the population.

What makes our strategy different is that we don't use the concept of evolutionary generations. We manage our set of solutions as a whole that continuously evolve over time. Allowing individuals of different ages to breed and generate offspring, as it happens in nature. One difference to highlight in our algorithm is that both the crossover (reproduction) and mutation (growth) are executed through independent processes that randomly affect the population. We display the general model concept for the Animal Life Cycle Algorithm (ALCA) in Figure 1.

In this research, we propose a new alternative for reproduction, a new breeding crossover approach for evolutionary algorithms using the classic one-point crossover as a reference to our proposal. We performed comparison experiments using the mathematical benchmark functions introduced in the Competition on Evolutionary Computation for the 2017 edition (CEC-2017) for evaluation. We finalize with a statistical Z-Test with a 95 percent confidence level to determine the best alternative.

We organized this paper in the following structure. First, we illustrate our alternative crossover proposal, from the inspiration to a detailed example. We compare the canonical one-point crossover with our proposed solution in section 2, followed by our experiment configuration and results in section 3, where we analyze and describe some of our research findings in section 4. We finalize by presenting some inferences based on the results of our experiments in section 5.

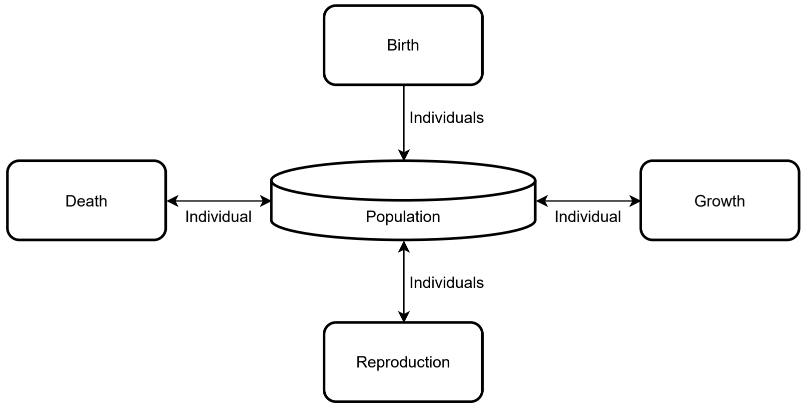

Fig. 1 Animal Life Cycle Algorithm model.

2 Proposal

Let's imagine that colors black and white fell in love and had beautiful children. If we had to guess what color their offspring would be, it could only be predictable its children's color would be inside the domain shown in Figure 2.

Fig. 2 Black and white offspring predictable color domain.

If we replace colors black and white with values 0 and 1 respectively, how many values can we find in between? We could replace attributes, switching from color to height, strength, agility, or any other feature we might need to focus on. The parents' attribute values will determine how their offspring's attributes are defined. This paper focuses on breeding (or reproduction), whereas in earlier versions of the algorithm, we used the traditional (GA) one-point crossover.

2.1 Crossover Proposal

In this paper, we propose a different alternative to the classical approach, where part of the genetic information is directly copied to each of the offspring in the crossover operator, where this type of crossover may not perform well in continuous optimization problems. We provide an example of the One Point Crossover in Figure 3.

ONE POINT CROSSOVER

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Parent A	-5.23	2.36	3.74	-4.52	9.1	-1.57	6.32	-4.68	0.15	-2.53
Parent B	-1.56	-3.91	7.92	-1.67	-3.72	-5.85	3.64	-1.15	2.64	0.23

Crossover Point 3

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Offspring A	-5.23	2.36	3.74	-1.67	-3.72	-5.85	3.64	-1.15	2.64	0.23
Offspring B	-1.56	-3.91	7.92	-4.52	9.1	-1.57	6.32	-4.68	0.15	-2.53

Fig. 3 Classical approach example for the One Point Crossover, with crossover point 3.

In this proposal, we use the parent's genetic information for each gene, using those values as lower and upper bounds of a range, where a random value within that range determines the new value for that gene index of the offspring. Our proposal example for the Continuous Range Crossover is displayed in Figure 4, where it should be noted that we construct a template that will be used for all the offspring generated by each couple.

CONTINUOUS RANGE CROSSOVER

	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10
Parent A	-5.23	2.36	3.74	-4.52	9.1	-1.57	6.32	-4.68	0.15	-2.53
Parent B	-1.56	-3.91	7.92	-1.67	-3.72	-5.85	3.64	-1.15	2.64	0.23

Random Value [MIN … MAX]

Offspring A

	MIN	MAX
D1	-5.23	-1.56
D2	-3.91	2.36
D3	3.74	7.92
D4	-4.52	-1.67
D5	-3.72	9.1
D6	-5.85	-1.57
D7	3.64	6.32
D8	-4.68	-1.15
D9	0.15	2.64
D10	-2.53	0.23

Offspring B

	MIN	MAX
D1	-5.23	-1.56
D2	-3.91	2.36
D3	3.74	7.92
D4	-4.52	-1.67
D5	-3.72	9.1
D6	-5.85	-1.57
D7	3.64	6.32
D8	-4.68	-1.15
D9	0.15	2.64
D10	-2.53	0.23

Fig. 4 Our proposal example for the Continuous Range Crossover mating the same couple of individuals.

3 Experiments

To validate our proposal, we performed experiments using the mathematical benchmark functions introduced in the Competition on Evolutionary Computation for the 2017 edition (CEC-2017) for evaluation to compare the traditional one-point crossover and our proposed strategy. We finalize with a statistical Z-Test with a ninety-five percent confidence level to determine the best alternative. Table 1 lists the thirty mathematical benchmark functions used for evaluation, to obtain the comparison results shown in our research.

Table 1 Mathematical benchmark functions introduced in the Competition on Evolutionary Computation for the 2017 edition (CEC-2017).

CEC-2017 Mathematical Benchmark Functions			
Fx	**Function Name**	**Fx**	**Function Name**
F1	Shifted and Rotated Bent Cigar Function	F16	Hybrid function 6
F2	Shifted and Rotated Sum of Different Power Function	F17	Hybrid function 7
F3	Shifted and Rotated Zakharov Function	F18	Hybrid function 8
F4	Shifted and Rotated Rosenbrock’s Function	F19	Hybrid function 9
F5	Shifted and Rotated Rastrigin’s Function	F20	Hybrid function 10
F6	Shifted and Rotated Schaffer F7 Function	F21	Composition function 1
F7	Shifted and Rotated Lunacek Bi-Rastrigin’s Function	F22	Composition function 2
F8	Shifted and Rotated Non-Continuous Rastrigin’s Function	F23	Composition function 3
F9	Shifted and Rotated Levy Function	F24	Composition function 4
F10	Shifted and Rotated Schwefel’s Function	F25	Composition function 5
F11	Hybrid function 1	F26	Composition function 6
F12	Hybrid function 2	F27	Composition function 7
F13	Hybrid function 3	F28	Composition function 8
F14	Hybrid function 4	F29	Composition function 9
F15	Hybrid function 5	F30	Composition function 10

3.1 Experimental Configuration

We used the Animal Life Cycle Algorithm (ALCA) for our experiments, switching between the breeding strategies (One Point and Continuous Range Crossover) with the same general setup. In Table 2, we can find the General Configuration used to run the experiments for our research, where all values from this configuration remained constant during all phases of the experiment runs.

For each of the thirty mathematical benchmark functions from the CEC-2017, we evaluated the ten and thirty dimensions, where we performed fifty-one runs per dimension. To find and determine the best alternative finalized with a statistical Z-Test with a ninety-five percent confidence level.

Table 2 Animal Life Cycle Algorithm (ALCA) General Configuration.

ALCA General Configuration	
Function name	**ALL**
Population	500
Evaluations	10,000 * Dim
Crossover rate	100
Mutation rate	7
Max age	5
Tournament rep.	100
Sample size	20
Base approval	80
Goal approval	200

3.2 Experimental Results

We describe how to interpret the results presented in our Box and Whisker charts for each mathematical function. We display the ten and thirty-dimension charts on the left and right sides. We can find the results accumulated in 51 runs, in green color the One Point crossover, and in blue color the Continuous Range crossover. The closer the error results get to the zero value, the better.

Fig. 5 Box and Whisker charts for mathematical functions F1 and F2. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

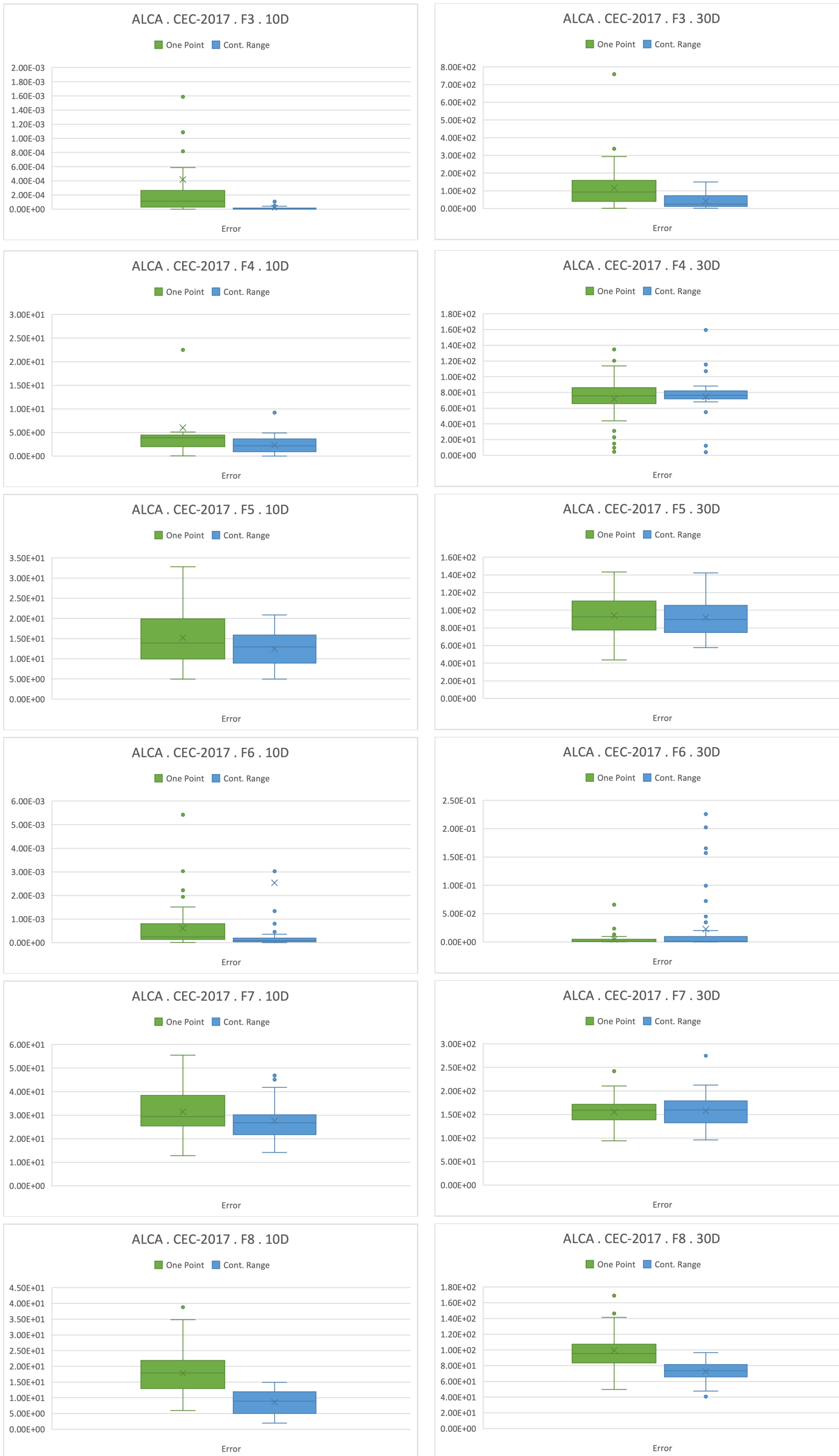

Fig. 6 Box and Whisker charts for mathematical functions F3 to F8. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

Fig. 7 Box and Whisker charts for mathematical functions F9 to F14. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

Fig. 8 Box and Whisker charts for mathematical functions F15 to F20. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

Fig. 9 Box and Whisker charts for mathematical functions F21 to F26. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

Fig. 10 Box and Whisker charts for mathematical functions F27 to F30. We display the ten and thirty-dimension charts on the left and right sides for the results accumulated in 51 runs.

4 Discussion

For each experiment, we ran fifty-one independent executions per CEC-2017 mathematical benchmark function and dimensions specified. We recorded the following results: best-found error average and standard deviation to calculate the statistical

Z-Test value. The labels used on the results of our summarized results tables are the following:

- Fx: is the mathematical benchmark function.
- Mean: is the best-found error average.
- St-Dev: is the standard deviation calculated from the error.
- Z: is the calculated statistical Z-Test value.

To find and determine the best alternative finalized with a statistical Z-Test with a ninety-five percent confidence level. Here are the results statistics values: Table 3 shows the condensed ALCA evaluation results for the CEC-2017 functions running on ten dimensions (D) for alternatives One Point and Continuous Range crossover; Table 4 displays the condensed ALCA evaluation results for the CEC-2017 mathematical operations running on thirty D for the same crossover alternatives.

Table 3 Condensed ALCA evaluation results for the CEC-2017 functions running on ten dimensions for alternatives One Point and Continuous Range crossover.

ALCA . CEC-2017 . 10D						
Crossover	One Point Xover			Cont. Range Xover		
Fx	Mean	Std. Dev.	Z	Mean	Std. Dev.	Z
F1	2.14E+03	2.14E+03	5.42E+00	4.43E+02	6.46E+02	**-5.42E+00**
F2	1.81E-03	2.63E-03	2.38E+00	8.54E-04	1.16E-03	**-2.38E+00**
F3	4.19E-04	1.16E-03	2.49E+00	1.33E-05	2.20E-05	**-2.49E+00**
F4	6.00E+00	1.31E+01	2.00E+00	2.31E+00	1.78E+00	**-2.00E+00**
F5	1.53E+01	7.06E+00	2.45E+00	1.24E+01	4.14E+00	**-2.45E+00**
F6	6.05E-04	9.22E-04	-1.43E+00	2.53E-03	9.61E-03	1.43E+00
F7	3.15E+01	9.06E+00	2.26E+00	2.76E+01	8.38E+00	**-2.26E+00**
F8	1.78E+01	7.38E+00	8.00E+00	8.62E+00	3.53E+00	**-8.00E+00**
F9	2.02E+01	2.24E+01	1.85E+00	1.25E+01	1.96E+01	**-1.85E+00**
F10	4.96E+02	2.26E+02	**-2.02E+00**	5.94E+02	2.63E+02	2.02E+00
F11	1.16E+01	6.01E+00	**-2.44E+00**	1.49E+01	7.48E+00	2.44E+00
F12	1.70E+04	1.99E+04	2.00E+00	1.10E+04	8.21E+03	**-2.00E+00**
F13	7.39E+03	8.16E+03	1.08E+00	5.94E+03	5.03E+03	-1.08E+00
F14	3.24E+02	6.05E+02	4.65E-01	2.67E+02	6.34E+02	-4.65E-01
F15	5.70E+02	1.31E+03	1.25E+00	3.07E+02	7.46E+02	-1.25E+00
F16	1.56E+02	1.28E+02	5.05E-01	1.43E+02	1.18E+02	-5.05E-01
F17	3.17E+01	3.65E+01	1.81E+00	2.11E+01	1.98E+01	**-1.81E+00**
F18	2.26E+03	2.88E+03	9.03E-01	1.73E+03	3.06E+03	-9.03E-01
F19	1.88E+03	3.24E+03	4.11E-01	1.63E+03	2.95E+03	-4.11E-01
F20	9.50E+00	8.32E+00	-1.35E+00	1.20E+01	9.92E+00	1.35E+00
F21	1.15E+02	3.86E+01	4.93E-02	1.15E+02	3.58E+01	-4.93E-02
F22	1.00E+02	2.07E+01	**-2.57E+00**	1.08E+02	8.70E+00	2.57E+00
F23	3.21E+02	7.31E+00	**-2.53E+00**	3.25E+02	8.91E+00	2.53E+00
F24	3.28E+02	8.06E+01	5.36E+00	2.15E+02	1.28E+02	**-5.36E+00**
F25	4.35E+02	2.61E+01	1.39E+00	4.28E+02	2.41E+01	-1.39E+00
F26	3.99E+02	1.25E+02	7.55E-01	3.82E+02	9.35E+01	-7.55E-01
F27	3.88E+02	1.03E+01	-5.43E-01	3.90E+02	1.13E+01	5.43E-01
F28	4.11E+02	7.92E+01	1.93E+00	3.80E+02	8.46E+01	**-1.93E+00**
F29	3.00E+02	4.32E+01	7.44E-01	2.94E+02	3.30E+01	-7.44E-01
F30	3.21E+02	1.52E+02	-7.14E-01	3.44E+02	1.81E+02	7.14E-01
Total (W)	**One Point Xover**		**4 of 30**	**Cont. Range Xover**		**12 of 30**

We performed the statistical Z-Test for both crossover alternatives with a 95 percent confidence level. When the One Point passes the test, the Z value is highlighted in bold on the left column. When the Continuous Range passes the statistical test, the Z value is highlighted in bold on the right column. Where we did not find enough evidence to declare one choice as the best alternative, no value was highlighted.

Table 4 Condensed ALCA evaluation results for the CEC-2017 functions running on thirty dimensions for alternatives One Point and Continuous Range crossover.

ALCA . CEC-2017 . 30D						
Crossover	One Point Xover			Cont. Range Xover		
Fx	Mean	Std. Dev.	Z	Mean	Std. Dev.	Z
F1	4.92E+03	6.21E+03	1.60E+00	3.35E+03	3.33E+03	-1.60E+00
F2	9.21E+06	4.77E+07	1.37E+00	6.96E+04	2.50E+05	-1.37E+00
F3	1.17E+02	1.18E+02	4.33E+00	4.16E+01	3.85E+01	**-4.33E+00**
F4	7.17E+01	3.14E+01	-5.29E-01	7.46E+01	2.52E+01	5.29E-01
F5	9.38E+01	2.41E+01	4.61E-01	9.18E+01	2.04E+01	-4.61E-01
F6	4.67E-03	1.02E-02	**-2.42E+00**	2.28E-02	5.26E-02	2.42E+00
F7	1.55E+02	3.04E+01	-3.88E-01	1.58E+02	3.44E+01	3.88E-01
F8	9.94E+01	2.43E+01	6.82E+00	7.29E+01	1.32E+01	**-6.82E+00**
F9	1.37E+03	7.21E+02	4.60E+00	8.19E+02	4.64E+02	**-4.60E+00**
F10	2.79E+03	5.59E+02	**-2.22E+00**	3.02E+03	4.63E+02	2.22E+00
F11	7.31E+01	2.72E+01	-1.09E+00	7.90E+01	2.75E+01	1.09E+00
F12	2.18E+05	1.94E+05	4.61E+00	8.73E+04	5.95E+04	**-4.61E+00**
F13	1.95E+04	2.73E+04	2.09E+00	1.10E+04	8.96E+03	**-2.09E+00**
F14	2.22E+04	2.14E+04	2.83E+00	1.27E+04	1.06E+04	**-2.83E+00**
F15	1.25E+04	1.41E+04	4.96E+00	2.37E+03	3.82E+03	**-4.96E+00**
F16	1.26E+03	3.45E+02	1.49E-01	1.26E+03	2.49E+02	-1.49E-01
F17	6.02E+02	1.76E+02	1.17E+00	5.54E+02	2.36E+02	-1.17E+00
F18	1.29E+05	1.17E+05	2.92E+00	7.44E+04	6.27E+04	**-2.92E+00**
F19	9.78E+03	1.16E+04	3.04E+00	4.38E+03	5.08E+03	**-3.04E+00**
F20	4.83E+02	1.69E+02	1.02E-01	4.79E+02	2.03E+02	-1.02E-01
F21	3.03E+02	2.44E+01	4.40E+00	2.85E+02	1.62E+01	**-4.40E+00**
F22	1.76E+03	1.72E+03	6.07E+00	2.16E+02	5.84E+02	**-6.07E+00**
F23	4.62E+02	3.06E+01	**-5.27E+00**	5.05E+02	4.97E+01	5.27E+00
F24	6.45E+02	7.43E+01	**-2.31E+00**	6.78E+02	7.03E+01	2.31E+00
F25	3.85E+02	1.49E+01	**-3.06E+00**	3.97E+02	2.40E+01	3.06E+00
F26	2.40E+03	4.88E+02	-1.23E+00	2.61E+03	1.17E+03	1.23E+00
F27	4.94E+02	1.23E+01	2.98E-01	4.93E+02	1.95E+01	-2.98E-01
F28	4.63E+02	4.29E+01	9.26E+00	3.98E+02	2.58E+01	**-9.26E+00**
F29	9.22E+02	2.36E+02	1.55E+00	8.56E+02	2.00E+02	-1.55E+00
F30	3.53E+03	4.15E+03	4.09E+00	1.08E+03	1.04E+03	**-4.09E+00**
Total (W)	**One Point Xover**		**5 of 30**	**Cont. Range Xover**		**13 of 30**

5 Conclusions

To validate our research, we performed the statistical Z-Test with a ninety-five percent confidence level to analyze the alternatives we present in this proposal. In the ten-dimension category, we found that One Point crossover was the better alternative on four occasions, while our Continuous Range proposal was better on twelve other functions out of thirty. For the thirty-dimensions category, the One Point crossover was better on five occasions, while the Continuous Range crossover was the best alternative on another thirteen. Our experiment results confirm that our solution can be considered a good alternative.

Some researchers might consider our proposal similar to what is used by algorithms such as Differential Evolution, where we think of our proposal as a variation of existing proposals because we expect this new operator to allow the offspring to continue exploring new search spaces with the birth of individuals. Experimental results indicate that our proposed operator may be a good alternative for the canonical One Point crossover.

Acknowledgements This paper has been supported in part by TecNM Project 15340.22-P.

References

1. Castillo, O., Valdez, F., Soria, J., Amador-Angulo, L., Ochoa, P., Peraza, C.: Comparative study in fuzzy controller optimization using bee colony, differential evolution, and harmony search algorithms. Algorithms 12(1), 9 (2019)
2. Valdez, F.: Swarm intelligence: A review of optimization algorithms based on animal behavior. Recent Advances of Hybrid Intelligent Systems Based on Soft Computing pp. 273–298 (2021)
3. Acherjee, B., Maity, D., Kuar, A.S.: Ultrasonic machining process optimization by cuckoo search and chicken swarm optimization algorithms. International Journal of Applied Metaheuristic Computing (IJAMC) 11(2), 1–26 (2020)
4. Porto, V.W.: Evolutionary programming. In: Evolutionary Computation 1, pp. 127–140. CRC Press (2018)
5. Back, T.: Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms. Oxford university press (1996)
6. Valdez, M.G., Guervós, J.J.M.: A container-based cloud-native architecture for the reproducible execution of multi-population optimization algorithms. Future Generation Computer Systems 116, 234–252 (2021)
7. García-Valdez, M., Trujillo, L., Merelo, J.J., de Vega, F.F., Olague, G.: The evospace model for pool-based evolutionary algorithms. Journal of Grid Computing 13(3), 329–349 (2015)
8. Merelo, J.J., García-Valdez, M., Castillo, P.A., García-Sánchez, P., Cuevas, P., Rico, N.: Nodio, a javascript framework for volunteer-based evolutionary algorithms: first results. arXiv preprint arXiv:1601.01607 (2016)
9. Felix-Saul, J.C., Valdez, M.G., Guervós, J.J.M.: A Novel Distributed Nature-Inspired Algorithm for Solving Optimization Problems, pp. 107–119. Springer International Publishing, Cham (2022)
10. Felix-Saul, J.C., Garcia Valdez, M.: Recovering from Population Extinction in the Animal Life Cycle Algorithm (ALCA), pp. 425–440. Springer Nature Switzerland, Cham (2023)
11. Read, K., Ashford, J.: A system of models for the life cycle of a biological organism. Biometrika 55(1), 211–221 (1968)
12. Holland, J.H.: Genetic algorithms. Scientific american 267(1), 66–73 (1992)
13. Holland, J.H.: Genetic algorithms and adaptation. Adaptive control of ill-defined systems pp. 317–333 (1984)