

THE EFFECTS OF CMA-ES STYLE SELECTION AND RESTART CRITERIA ON DE

Mark Wineberg and Samuel Opawale

University of Guelph
School of Computer Science
50 Stone Road East Guelph, Ontario, N1G 2W1
Canada
mwineber@uoguelph.ca, sopawale@uoguelph.ca

Abstract: Over the years, a lot of research has gone into the creation of different mutation operators and adaptive parameters for differential evolution (DE). However, the literature is fairly quiet about automatically setting population size and completely silent about varying the selection operator used within DE.

In this paper, we steal a page from CMA-ES/IPOP: using ES-style $\mu + \lambda$ selection, which selects across the entire population, in place of more individualistic DE selection with its use of local selection on the target and its child. We find that the most effective choice of selection can depend on the function being optimized, although for most of the functions we tested, the original DE selection was preferable. When adding IPOP style restarting, EqualFunValHist is the most applicable of the stagnation criteria, and it is used to trigger the doubling of the population size upon restart. The initial population size is set to the same as CMA-ES. Here we find, that the restartable DE behave as well and better as regular DE with population size set as lower than the default settings used.

Keywords: CMA-ES, DE, selection, restart, IPOP, ES.

1 Introduction

Differential evolution (DE) and Covariance matrix adaptation evolutionary strategies (CMA-ES) have proven to be one of the most powerful forms of evolutionary algorithms (EA). EA is a population based algorithm that uses biologically inspired operators in solving optimization problems. Both Algorithms act on real numbers in order to provide solutions to a multidimensional and non-differentiable problems.

As proposed in 1996 by Hansen and Ostermeier, the idea of CMA-ES is having a system that will enable the evolution strategy (ES) to adapt to the correct scaling of a given problem and also ensure invariance with respect to any rotation of the fitness function [4]. The goal of CMA-ES, which has shown great success on benchmark functions, is to fit the multivariate normal distribution of ES mutations to the contour of the objective functions [10]. DE on the other hand is a typical example of a population based algorithm that is designed to optimize functions in an n-dimensional continuous domain [6, 11].

Though CMA-ES and DE are applied to similar problems, the way they approach evolution when producing their offspring appears vastly different. CMA-ES creates a centralized model, then throws away the old population and generates a new one from its model de novo, while DE produce an offspring directly by combining individuals from the previous population. While very different in character, both algorithms can be analyzed under a common framework: each generates samples that are evaluated and undergo selection to change where the algorithm will search next. In EC, researchers over the years have tried to compare these algorithms head-to-head over different test functions, which shows what the algorithm is doing without any knowledge of why the algorithm is behaving the way it is.

Black box optimization benchmark (BBOB) is an area of specialization in EC that deals with providing solutions to black box problems. We call the sole use of analyzing EAs using only a test width of black-box problem black-box experimental design (BBED). Unfortunately, BBED is limited to looking at the performance of EAs without any internal knowledge of their workings. BBED tends to be more efficient over the years in terms of performance because it allows different EAs to be tested on different functions. Ultimately, BBED is that it tells us “what” the algorithm is doing not “why”.

White-box experimental design (WBED) can be defined as a method that focuses on the internal workings of an algorithm. It is a means of understanding the behavior of an algorithm through careful manipulation and

experimentation of the various component parts of the algorithm, often through comparisons of the internal workings of similar algorithms, including the removal, addition, swapping, etc. of component segments of the algorithms.

We will begin the investigation of the internal structures of these algorithms, not only to provide a better understanding of them, but to also create an enabling environment for taking a critical look at why CMA-ES is performing better than DE. Breaking down the steps of the EA in each case (Initialization, selection, mutation, recombination, crossover and evaluation), isolating them and exploring them in controlled experiments will help in understanding the similarities and differences between them. In this paper, we begin this investigation by looking at the difference in selection techniques and automated population size determination (as done in IPOP) to see if these are the main explanatory factors in explaining the performance difference between the two algorithms

The structure of this paper is as follows: Section III compares CMA-ES and DE while section IV shows our approach in getting these two-algorithm close to each other. Section relates us to our experimental design and set up and the results is discussed in section V. We conclude our thought and proposed possible future work in section VI.

2 Comparing and Contrasting CMA-ES and DE

2.1 CMA-ES and DE variants

There are different variants of CMA-ES such as LS-CMA-ES(Least-Square) [3], (1 + 1)-Cholesky-CMA-ES [5]. For this paper, we will concentrate on "LR-CMA-ES" and "IPOP".

LR-CMA-ES: Local Restart (LR CMA-ES) was proposed by Auger and Hansen in 2005 [1]. It is a quasi-parameter free, comparatively simple global optimization algorithm exploiting the advanced local search properties of the $(\mu/\mu_W, \lambda)$ -CMA-ES. IPOP-CMA-ES: Auger and Hansen further improved LR-CMA-ES by proposing a restart CMA-ES with increasing population IPOP-CMA-ES [1, 2]. Whenever a run of the $(\mu/\mu_W, \lambda)$ - CMA-ES is terminated due to a local stopping/termination criterion, the population size is increased by a factor η for the next run of the $(\mu/\mu_W, \lambda)$ - CMA-ES.

In DE, the process of mutation is the most common demarcation of one form of DE from another. Storn and Prince [7] listed many types of mutation that are still in common use today, they include:

$$\text{"DE/best/1 :"} \vec{V}_{i,G} = \vec{X}_{best,G} + F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) \tag{1}$$

$$\begin{aligned} \text{"DE/target - to - best/1 :"} \vec{V}_{i,G} = & \vec{X}_{i,G} + \\ & F.(\vec{X}_{best,G} - \vec{X}_{i,G}) \\ & + F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) \end{aligned} \tag{2}$$

$$\begin{aligned} \text{"DE/best/2 :"} \vec{V}_{i,G} = & \vec{X}_{best,G} + \\ & F.(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) \\ & + F.(\vec{X}_{r_3^i,G} - \vec{X}_{r_4^i,G}) \end{aligned} \tag{3}$$

$$\begin{aligned} \text{"DE/rand/2 :"} \vec{V}_{i,G} = & \vec{X}_{r_1^i,G} + \\ & F.(\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G}) \\ & + F.(\vec{X}_{r_4^i,G} - \vec{X}_{r_5^i,G}) \end{aligned} \tag{4}$$

The indices $r_1^i, r_2^i, r_3^i, r_4^i$ and r_5^i are mutually exclusive integers randomly chosen from range [1, NP], and all are different from the base index i. These indices are randomly generated once for each donor vector.

2.2 Differences and similarities between DE and CMA-ES

Broad brush-stroke differences

CMA-ES uses a form of central model reproduction in generating offspring while DE combine different individuals in generating offspring directly or by obeying the crossover rate condition depending on the type of DE

being used. One of the unique features of CMA-ES is the path length(P_c and P_σ). P_c is used in updating the covariance matrix C while P_σ is used in updating the global step size σ for the next generation but there is nothing like this in DE.

Finally, CMA-ES and DE use different terminologies when performing some of their operations e.g. crossover, which is used in DE versus recombination as used in CMA-ES.

Broad brush-stroke similarities

Selection in DE (Local elitism) [9] and selection in $(\mu + \lambda)$ CMA-ES is very similar to a form of selection in GA called tournament selection. The only difference between these selection type is that; tournament selection is used to select for reproduction while local elitism and global elitism is used to select for the next generation.

Also, both can be thought of as producing “samples” for search space. DE achieved this by performing a difference operator to add “noise” of a step-size while CMA-ES explicitly adds “Gaussian noise” to a center (instead of individuals) to produce the sample but noise is still added in other to create a new offspring/samples.

3 Selection Methods in CMA-ES and DE

Typically, the ES (μ, λ) selection, also known as truncation selection, is used in CMA-ES. This choice of selection allows the best offspring to be selected for the next generation and the rest of the population is deleted. $(\mu + \lambda)$ CMA-ES with elitism is the recent form of CMA-ES proposed in 2017 by van Rijn [8]. This method allows us to add the offspring to the original population before selecting the best fittest individuals for the next generation. From [8], it can be seen that $(\mu + \lambda)$ CMA-ES often performs better and faster than (μ, λ) CMA-ES.

DE on the other hands selects based on the individual called the target, going into a mini tournament with a single offspring produced from it. Consequently, this selection operate at a local individual level while $(\mu + \lambda)$ CMA-ES operates at a global population wise level. Both selection techniques implicitly allow the best parent from the previous generation into the next analogous to elitism used in genetic algorithms, i.e. DE implements a form of *local elitism* while CMA-ES implements a form of *global elitism*. From-here-on-in, we will refer to the DE selection as *local elitist selection* and the $(\mu + \lambda)$ CMA-ES selection as *global elitist selection*.

Both local and global elitist selection are similar to tournament selection as used in the GA. The only difference is that tournament selection is used to select members of the population for reproduction while local and global elitist selection is used after reproduction has been completed and is used to change the composition of and proportionality within the population.

Now that is it clear that both DE and CMA-ES use comparable selection techniques, it becomes possible to see the effects of the different selection techniques on their respective algorithms by swapping them. This can be done in a straightforwardly for DE by combining all targets with all their offspring and performing global elitist selection instead of local elitist selection.

Using local elitist selection with CMA-ES is more challenging. Since CMA-ES uses a centralized model when producing offspring, locality is destroyed. Each member of the offspring is not produced by any specific parent but through an amalgamation of all parents culminating in the center from which the offspring are produced.

However, a form of locality can be reconstituted by finding for each offspring, the closest parent using Euclidean distance, which we call *matching*. To make it as close as possible to DE, each offspring must be matched to only one parent. This leads to ambiguities that arise when two offspring find that they are close to the same parent. Consequently, the order of matching could make a difference if a parent is awarded to one close offspring, it is no longer available to the second one who would then choose other parents, which in turn could affect the matching of later offspring.

In our current implementation, we match offspring in order of their fitness with the more fit offspring matched first. There are many possible approaches to match ordering e.g. the less fit is matched earlier or the matching is done using parent fitness to do the ordering.

Preliminary experiments on CMA-ES with local elitist selection did not prove deleterious across our entire current test suit. Consequently, there is no need to present a detailed analysis and for the experimentation section, we will concentrate on DE and compare global versus local elitist selection on it. This will give us

insight into the behavior of both the selection operator and the nature of DE itself and possibly indicate where and why local elitist selection is difficult or performs poorly within CMA-ES.

4 Experimental Design

This section provides us with the detailed information about our approach to this paper. The first preliminary work done is to control and isolate factors as should be done in proper experimental design and consequently focusing our attention on DE and CMA-ES/ IPOPOP systems instead of their modern incarnations which add feature after feature. Once basic understanding is achieved, more advance features can be added in future work. For this study, we begin by adding global elitist selection to DE.

The next method we used in bringing DE close to ES is to add an increasing population (IPOPOP) style stagnation detection with restart (see [1, 2] for details). Most of the "restart criteria" are explicitly about the covariance matrix, step-size or path histories used in CMA-ES model. Since we are explicitly not using the centralized model in this study, some of the restart criteria will not be included here (although some could be modified to fit with a lot of messaging e.g. a covariance matrix can be empirically calculated from the DE population and then undergo a Eigen decomposition to apply NoEffectAxis, NoEffectCoord or ConditionCOV) and this leaves us with only "EqualFunValHist" which can be easily applied to DE. The default is to double the population size when a restart is called, which we kept the same.

For our initial experiment, we perform a fully factorial design which includes the following factors: five fitness functions (Ackley, Rastrigin, Levy, Griewank and elliptical), two number of dimensions (25, 50) against the four different systems either with or without elitism or with or without centre. Each treatment was repeated 60 times.

The experiment for this design is set up by considering two of the different types of mutation discussed in equation (1) and (3); classic and best. These choices of mutations are allowed to run with both global and local elitist selection over 30 reps in order to see if the choice of selection will have effect on the performance of DE. The concept of restart in CMA-ES was also introduced into DE with EquiunvalHist as the only stopping criteria. DE with restart will be test against ordinary restart to see its effect on the entire system.

The fitness functions used are now detailed:

1. Elliptical:

$$f(x) = \sum_{i=1}^d \sum_{j=1}^i x_j^2 \quad (5)$$

2. Ackley:

$$f(x) = -a \exp\left(-b \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d} \sum_{i=1}^d \cos(cx_i)\right) + a + \exp(1) \quad (6)$$

where $a = 20$, $b = 0.2$ and $c = 2\pi$

3. Griewank:

$$f(x) = \sum_{i=1}^d \frac{x_i^2}{4000} - \pi_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (7)$$

4. Rastrigin:

$$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10\cos(2\pi x_i)] \quad (8)$$

5. Levy:

$$f(x) = \sin^2(\pi\omega_1) + \sum_{i=1}^{d-1} (\omega_i - 1)^2 [1 + 10\sin^2(\pi\omega_i + 1)] + (\omega_d - 1)^2 [1 + \sin^2(2\pi\omega_d)] \quad (9)$$

where $\omega_i = 1 + \frac{x_i - 1}{4}$, for all $i = 1, \dots, d$

Each of the above listed fitness function is allowed to run with angle of rotation 0 deg and 5 deg respectively.

Table 1: Results from the full ANOVA model (all factor main effects and interaction included)

Source	df	Sums of Squares	Mean Square	F-ratio	P-value
Intercept	1	21745	21745	8398135	< 0.0001
fn	3	25.8359	8.61195	3326	< 0.0001
deg	1	0.163783	0.163783	63.254	< 0.0001
fn*deg	3	0.272964	0.090988	35.14	< 0.0001
tmp	1	0.003982	0.003982	1.5379	0.2151
fn*tmp	3	0.031205	0.010402	4.0173	0.0073
deg*tmp	1	0.002074	0.002074	0.8009	0.3709
fn*deg*tmp	3	0.006011	0.002004	0.77386	0.5086
sel	1	0.030895	0.030895	11.932	0.0006
fn*sel	3	0.01263	0.00421	1.6259	0.1814
deg*sel	1	0.023831	0.023831	9.2038	0.0024
fn*deg*sel	3	0.082399	0.027466	10.608	< 0.0001
tmp*sel	1	0.010681	0.010681	4.1252	0.0424
fn*tmp*sel	3	0.00571	0.001903	0.73514	0.531
deg*tmp*sel	1	0.002732	0.002732	1.0551	0.3045
fn*deg*tmp*sel	3	0.013544	0.004515	1.7436	0.1561
rst	1	0.268592	0.268592	103.73	< 0.0001
fn*rst	3	0.551727	0.183909	71.027	< 0.0001
deg*rst	1	0.001851	0.001851	0.71481	0.398
fn*deg*rst	3	0.030341	0.010114	3.906	0.0085
tmp*rst	1	0.011864	0.011864	4.582	0.0324
fn*tmp*rst	3	0.03706	0.012353	4.7709	0.0026
deg*tmp*rst	1	0.002875	0.002875	1.1102	0.2922
fn*deg*tmp*rst	3	0.002733	0.000911	0.35182	0.7878
sel*rst	1	0.280223	0.280223	108.22	< 0.0001
fn*sel*rst	3	1.34556	0.448519	173.22	< 0.0001
deg*sel*rst	1	0.000163	0.000163	0.06284	0.8021
fn*deg*sel*rst	3	0.083803	0.027934	10.789	< 0.0001
tmp*sel*rst	1	0.000696	0.000696	0.26898	0.6041
fn*tmp*sel*rst	3	0.012666	0.004222	1.6306	0.1803
deg*tmp*sel*rst	1	0.000158	0.000158	0.06118	0.8047
fn*deg*tmp*sel*rst	3	0.00581	0.001937	0.74792	0.5235
Error	1856	4.80568	0.002589		
Total	1919	33.9401			

Table 2: Legend of factor used in the ANOVA table

	Legend	Values
fn	Function	[Griewank, Elliptical, Levy, Rastrigin]
deg	Degree of Fn Rotation	[0, 5]
mut	Mutation Type	[Classic, Best]
sel	Selection Type	[DE, Trunc]
rst	Restartable	[T, F]

5 Results, Discussion and Conclusion

To analyze the results of the factorial design detailed above, a multi-way ANOVA was used. A normal distribution plot was performed on the residuals from the ANOVA, and it was determined to be highly non-normal. The typical technique for performing a non-parametric ANOVA to handle non-normal data is either Kruskal- Wallis or Friedman. Unfortunately, both only perform a one way and not a multi-way ANOVA as needed here. As indicated in [12], a Box-Cox transform can often turn non-normal EC results to become normally distributed. The Box-Cox transform is:

$y' = \frac{y^p - 1}{p}$, where y is the number of evaluation a run took before terminating (either by finding a solution or hitting max evaluation). Following this procedure, we manually explored a variety of Box-Cox p settings to maximize the r^2 measure that evaluates the fit of a regression line through the normal-distribution plot. We found that a Box-Cox p setting of -0.224 produced the most normal result with an $r^2 = 97.4\%$, which is highly normally distributed. All pair-wise comparisons done using the ANOVA, have the Sheffe post-hoc correction applied.

The results can be seen in Table(s) 1-3. The first table gives the result of the full ANOVA model including all the interactions, from which we can examine which factors are statistically significant and which are irrelevant. We use a confidence level of 99% eliminating from the model any factor that has an associated p-value greater than 0.01. In some cases, it may seem as if we are deleting factors that have p-values less than 0.01 and leaving in factors greater than 0.01. This is because when the model changes, the Box-Cox p setting needs to be changed as well. This improves the normality of the residuals but also changes the p-value of the ANOVA

Table 3: Results from the restricted ANOVA model where non-significant factor turns have been removed (using a confidence level of 99%)

Source	df	Sums of Squares	Mean Square	F-ratio	P-value
Intercept	1	21745	21745	8332605	< 0.0001
fn	3	25.8359	8.61195	3300.1	< 0.0001
deg	1	0.163783	0.163783	62.761	< 0.0001
fn*deg	3	0.272964	0.090988	34.866	< 0.0001
fn*mut	3	0.031205	0.010402	3.9859	0.0077
sel	1	0.030895	0.030895	11.839	0.0006
deg*sel	1	0.023831	0.023831	9.132	0.0025
fn*deg*sel	3	0.082399	0.027466	10.525	< 0.0001
rst	1	0.268592	0.268592	102.92	< 0.0001
fn*rst	3	0.551727	0.183909	70.473	< 0.0001
fn*mut*rst	3	0.03706	0.012353	4.7337	0.0027
sel*rst	1	0.280223	0.280223	107.38	< 0.0001
fn*sel*rst	3	1.34556	0.448519	171.87	< 0.0001
fn*deg*sel*rst	3	0.083803	0.027934	10.704	< 0.0001
Error	1890	4.9322	0.00261		
Total	1919	33.9401			

Table 4: The effects of the selection type used across different fitness function (with an without rotation) when looking at the number of evaluation to finding the solutions (lower values are better)

fn	deg	selection	LB	Avg	UB
elliptical	x	0 x local	2,228.1	2,779.2	3,515.5
elliptical	x	0 x global	2,713.2	3,004.8	3,337.4
elliptical	x	5 x local	2,120.1	2,633.3	3,315.1
elliptical	x	5 x global	2,668.6	2,975.2	3,328.1
griewank	x	0 x local	50,664.0	65,798.9	87,169.8
griewank	x	0 x global	59,289.2	72,408.7	89,446.5
griewank	x	5 x local	92,919.8	97,924.2	103,276.0
griewank	x	5 x global	96,786.3	98,956.9	101,189.7
levy	x	0 x local	1,675.0	2,108.3	2,694.5
levy	x	0 x global	2,088.4	2,350.2	2,655.1
levy	x	5 x local	1,698.9	2,147.3	2,757.3
levy	x	5 x global	2,184.8	2,429.3	2,709.6
rastrigin	x	0 x local	6,956.2	8,046.1	9,362.5
rastrigin	x	0 x global	8,280.7	11,932.6	17,897.6
rastrigin	x	5 x local	18,563.9	25,492.8	36,064.8
rastrigin	x	5 x global	12,130.8	17,351.9	25,790.7

Table 5: The effect of the ability to restart DE across different fitness function, when looking at the number of evaluation to finding the solutions (lower values are better)

fn	restart	LB	Avg	UB
elliptical	x F	3,494.2	3,692.0	3,904.2
elliptical	x T	1,979.2	2,227.1	2,515.8
griewank	x F	68,949.2	79,064.8	91,142.4
griewank	x T	78,285.7	85,206.3	92,922.7
levy	x F	2,679.3	2,838.4	3,009.8
levy	x T	1,589.2	1,813.9	2,080.6
rastrigin	x F	10,106.1	12,595.2	15,917.6
rastrigin	x T	13,004.0	15,621.3	18,945.7

Table 6: The interaction between selection type used and the ability to restart DE across different fitness function when looking at the number of evaluation to finding the solutions (lower values are better)

fn	selection	restart	LB	Avg	UB
elliptical	x local	x F	4,833.0	4,911.8	4,992.2
elliptical	x local	x T	1,369.4	1,618.3	1,927.7
elliptical	x global	x F	2,776.8	2,832.4	2,889.4
elliptical	x global	x T	2,726.6	3,158.9	3,682.1
griewank	x local	x F	50,583.7	65,562.4	86,654.1
griewank	x local	x T	92,548.6	98,318.1	104,552.1
griewank	x global	x F	84,498.1	96,304.4	110,287.1
griewank	x global	x T	62,821.7	74,252.9	88,460.9
levy	x local	x F	3,743.9	3,820.7	3,899.5
levy	x local	x T	1,054.5	1,283.6	1,579.9
levy	x global	x F	2,104.3	2,155.7	2,208.6
levy	x global	x T	2,278.4	2,656.1	3,117.2
rastrigin	x local	x F	8,387.3	9,620.2	11,092.7
rastrigin	x local	x T	13,959.7	20,238.0	30,581.9
rastrigin	x global	x F	10,315.3	16,843.0	29,643.2
rastrigin	x global	x T	10,163.4	12,262.1	14,943.3

making some factors more statistically significant and other factors less so. We remove the factors in order of largest p-value modify the Box-Cox p setting accordingly and repeat. The ANOVA model and results from this process is what is presented in Table 3.

We can see that almost all higher order interaction is not significant with the exception of function, degree, selection, and restart. This make sense as a rotated function can be thought of as a different fitness function and so would be tied together as an extended function set and so we still have really two-way interaction between different functions and selection operator used.

We will now move on from the general ANOVA summary to go into detail on the factor level pair-wise comparisons. To facilitate understanding the results found in Table(s) 4-6, confidence intervals are provided around the average number of evaluations used to reach the solution. These confidence intervals use a 99% confidence level and the appropriate Bonferonni correction and are only provided to give a sense of how wide or narrow the variance is around the average. The Bonferonni correction is overly harsh so the reader should not use confidence interval overlaps when comparing averages. Instead, all conclusion drawn below are based on pair-wise comparisons (p-values less then 0.01) after a Sheffe post-hoc correction has been applied. Assume all difference are statistically significant unless we specifically note otherwise.

First, we note that rotating the function by 5 degree seems to make the problem more difficult to solve as expected (not shown for space considerations). The sole exception is the elliptical function, where the change in rotation is not statistically significant. This makes sense, as the elliptical function is rotationally invariant when rotated around the origin.

When looking at the effectiveness of restart, we see from Table 5 that we have found the solution faster when using restart than a constant population size of 30 for both the elliptical and Levy problems. Looking at the ANOVA comparison p-values (not presented here for space considerations), there is no effect on restarting versus not restarting on Griewank. This is because the optimum was rarely if ever found for Griewank within the 100,000 evaluations allowed before cutoff. Also looking at the ANOVA comparison p-values, we saw that for Rastrigin, there was no statistical difference when classic mutation was used, but restart actually performed worst when best mutation was used. This can occur for a variety of reasons: if the population size is a priori optimally set right for the optimization problem, restarting has to build up to the correct population size while the fixed population size is already there. For Rastrigin with 5 dimensions, it appears that a population size of 30 is optimal.

Turning our attention to the selection type used, seen in Table 4, a deep look at the p-value produce through the ANOVA shows that there is no statistical difference between local and global elitist selection. However, for rastrigin with no rotation of the function local elitist selection prove better while with 5 deg rotation, global elitist selection won out.

Finally, there is an interaction between the type of selection used and whether we use the restarting capability or not as can be seen in Table 6. There is no statistical significance between global and local elitism under Griewank when restart is used. For elliptical, restart is better for local but there is no statistical significant for global. For Griewank and Rastrigin, restart is better for global but not for local. Restart is better for local with Levy and not restarting is better for global which is the opposite of Griewank. On local, Levy and Elliptical function the same, but on global Levy is reverse but no statistical difference in global. Consequently, if using restart with elliptical, use global elitist selection otherwise use local. For Griewank, if you are not using restart use local; if using restart there is little difference. For Levy, if using restart use local, if not using restart use global. For Rastrigin, if using restart use global, if not using restart use local.

In conclusion, for DE, global elitist selection can work better than local in some cases, but there seems to be a strong preference in the DE algorithm for the use of local elitist selection, as traditionally is done. This is in opposition to CMA-ES where global selection is the clear winner, although this is complicated by the artificiality of how we created “space” for local selection through matching parents with offspring using genetic closeness. Borrowing the “restart” facilities of IPOP proved more generally useful, with DE being able to relatively efficiently find an effective population size without the user having to “guess” what an appropriate population size might be.

References

- [1] Auger, A., Hansen, N.: Performance evaluation of an advanced local search evolutionary algorithm. In: *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 2, pp. 1777–1784. IEEE (2005)
- [2] Auger, A., Hansen, N.: A restart cma evolution strategy with increasing population size. In: *Evolutionary Computation, 2005. The 2005 IEEE Congress on*, vol. 2, pp. 1769–1776. IEEE (2005)
- [3] Auger, A., Schoenauer, M., Vanhaecke, N.: Ls-cma-es: A second-order algorithm for covariance matrix adaptation. In: *International Conference on Parallel Problem Solving from Nature*, pp. 182–191. Springer (2004)
- [4] Hansen, N., Ostermeier, A.: Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In: *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pp. 312–317. IEEE (1996)
- [5] Igel, C., Suttorp, T., Hansen, N.: A computational efficient covariance matrix update and a $(1+1)$ -cma for evolution strategies. In: *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pp. 453–460. ACM (2006)
- [6] Price, K.V.: Differential evolution: a fast and simple numerical optimizer. In: *Fuzzy Information Processing Society, 1996. NAFIPS., 1996 Biennial Conference of the North American*, pp. 524–527. IEEE (1996)
- [7] Price, K.V., Storn, R.M., Lampinen, J.A.: The differential evolution algorithm. *Differential evolution: a practical approach to global optimization* pp. 37–134 (2005)
- [8] van Rijn, S., Wang, H., van Stein, B., Bäck, T.: Algorithm configuration data mining for cma evolution strategies. In: *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 737–744. ACM (2017)
- [9] Sarma, J., De Jong, K.A.: An analysis of local selection algorithms in a spatially structured evolutionary algorithm. In: *ICGA*, pp. 181–187. Citeseer (1997)
- [10] Simon, D.: *Evolutionary optimization algorithms*. John Wiley & Sons (2013)
- [11] Storn, R., Price, K.: Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization* **11**(4), 341–359 (1997)
- [12] Wineberg, L.: Reexpressing problematic optimization data. In: *Proceedings of the 2017 Annual Conference on Genetic and Evolutionary Computation*, pp. 897–904. ACM (2017)