Differential Evolution and Non-separability: Using selective pressure to focus search

Andrew M. Sutton Monte Lunacek L. Darrell Whitley

Department of Computer Science
Colorado State University
Fort Collins, CO 80523
{sutton,lunacek,whitley}@cs.colostate.edu

ABSTRACT

Recent results show that the Differential Evolution algorithm has significant difficulty on functions that are not linearly separable. On such functions, the algorithm must rely primarily on its differential mutation procedure which, unlike its recombination strategy, is rotationally invariant. We conjecture that this mutation strategy lacks sufficient selective pressure when appointing parent and donor vectors to have satisfactory exploitative power on non-separable functions. We find that imposing pressure in the form of rank-based differential mutation results in a significant improvement of exploitation on rotated benchmarks.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search; G.1.6 [**Numerical Analysis**]: Optimization—*Global Optimization*

General Terms

Performance

Keywords

Differential Evolution, Selective Pressure

1. INTRODUCTION

Differential Evolution (DE) is a parallel direct search that evolves a fixed set of parameter vectors to optimize real valued functions. In contrast to Evolution Strategies (ES), DE does not maintain an explicit mutation distribution, but produces mutant vectors by sampling from scaled difference vectors between individuals. This procedure, called *differential mutation*, attempts to implicitly "shape" the mutation to model the ideal distribution for the population [8].

DE achieves "recombination" by allowing an individual to inherit a subset of parameter values from a mutant vector. This describes a potential hyper-rectangle, with the individual and its mutant vector in opposing corners, on which the new individual will lie. Of course, the sides of this hyper-rectangle are oriented with the coordinate axes, making search steps generated by crossover orthogonal to the function's coordinate directions. This operation is controlled by the crossover rate value CR which specifies the expected number of parameters inherited from the mutant vector. Essentially, CR determines the expected dimension of the subspace in which the offspring will lie.

Recent results [10, 3] suggest that DE struggles with functions that are not linearly separable. We conjecture that DE's bias toward separability arises from inefficient exploitation during the differential mutation step. In this paper we will explore the following two hypotheses.

1. When CR is low, DE can exploit the separability of a function by relying on crossover to produce axis-orthogonal steps. On non-separable functions, it must depend more directly on differential mutation.

2. DE lacks adequate selective pressure in the differential mutation step to make efficient progress on non-separable functions.

On functions with significant interaction between parameters, the dimension of the subspace in which the offspring lies must be larger in order to produce rotationally invariant steps. In this case, progress will depend more on the efficacy of the differential mutation procedure. Bäck [1] points out that methods depending mostly on recombination tend to perform quite well with weak selective pressure, whereas mutation-only methods such as ESs and Evolutionary Programming (EP) typically employ a strong selective pressure. DE, however, generates relatively weak selective pressure resulting from its unbiased mechanism for selecting parent vectors.

Our main focus is to identify factors that are causing DE's poor performance on non-separable functions. We will test our hypotheses by 1) Controlling crossover and mutation on rotated and non-rotated benchmarks and 2) Controlling selection pressure on vectors involved in differential mutation with a rank-based selection mechanism on rotated and non-rotated benchmarks. In Section 2 we introduce the Differential Evolution algorithm and discuss our method for imposing selective pressure on the differential mutation step. In Section 3 we present our empirical results. In Section 4 we make observations and discuss the implications of our results. We conclude the paper in Section 5.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GECCO'07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...$5.00.

2. DIFFERENTIAL EVOLUTION

For this research we use "classic DE", or the rand/1/bin strategy originally developed by Kenneth Price and Rainer Storn [9, 12]. To optimize a function in $\mathbb{R}^n$, classic DE keeps a population of λ vectors of dimension n. During a generation G, each vector $\mathbf{x}_i, i = 1, \ldots, \lambda$ has an opportunity to undergo differential mutation and crossover. Differential mutation is performed by selecting a *donor* vector $\mathbf{x}_{r_1}$, and two *parent* vectors $\mathbf{x}_{r_2}$ and $\mathbf{x}_{r_3}$. It is important that these vectors are disjoint, which we can enforce by having $i \neq r_1 \neq r_2 \neq r_3$. The mutant vector $\mathbf{v}_i$ for the i^{th} population vector is then generated by adding the scaled differential vector obtained from the two parent vectors to the donor vector.

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$$

Here, F is typically referred to as a *mutation scale factor* and is a positive real number less than 1.

Next, the mutant vector $\mathbf{v}_i$ and the original population vector $\mathbf{x}_i$ are recombined by selecting subsets of their components to include in a trial vector $\mathbf{u}_i$.

$$\mathbf{u}_{ij} \quad = \quad \begin{cases} \mathbf{v}_{ij} & \text{if } U(0,1) \leq CR \text{ or } j = j' \\ \mathbf{x}_{ij} & \text{otherwise} \end{cases}$$

for $j = 1, \ldots, n$ and $j' \in \{1, \ldots, n\}$. Here CR is the *crossover constant* and j' is a random integer that ensures the trial vector differs by at least one parameter. $U(0, 1)$ denotes a variate drawn from a uniform distribution on the range $[0, 1)$.

Finally, each trial vector $\mathbf{u}_i$ is compared with the original target population vector for replacement. Let $\mathbf{x'}_i$ denote the i^{th} population vector in generation $G + 1$. Let f be the fitness function (assuming minimization). Replacement is calculated as follows.

$$\mathbf{x'}_i = \begin{cases} \mathbf{u}_i & \text{if } f(\mathbf{u}_i) \leq f(\mathbf{x}_i) \\ \mathbf{x}_i & \text{otherwise} \end{cases}$$

Using this technique, DE essentially performs a parallel local search, and expands neighbors based on information sampled from the entire population. Each vector $\mathbf{x}_i$ must only compete with its own offspring $\mathbf{u}_i$. In fact, the *generational* selective pressure exists only between a given vector and its offspring produced, and not across the entire population.

The crossover strategy allows each population member to change a random subset of parameters individually. The CR parameter controls how many parameters, in expectation, are changed. For low values of CR, a small number of parameters are changed in each generation, and stepwise movement tends to be orthogonal to the coordinate axes. On the other hand, high values of CR cause most of the directions of the mutant vector to be inherited. Setting $CR = 1$, for instance, results in a mutation-only strategy. Figure 1 illustrates this effect by showing, for four values of CR, an empirical distribution of candidate trial vectors from a particular population.

A low CR value (e.g. 0 or 0.1) results in a search that changes each direction (or a small subset of directions) separately. In other words, the dimension of the subspace in which the offspring is produced will be low. This is an effective strategy for functions that are *separable* or *decomposable*, i.e. $f(\mathbf{x}) = \sum_i^n f_i(x_i)$. However, this approach will fail on *non-separable* or *epistatic* functions in which two or more parameters interact since parameters must be changed simultaneously to make progress [11]. Hansen and Kern [3] present empirical evidence that DE struggles with non-separability.

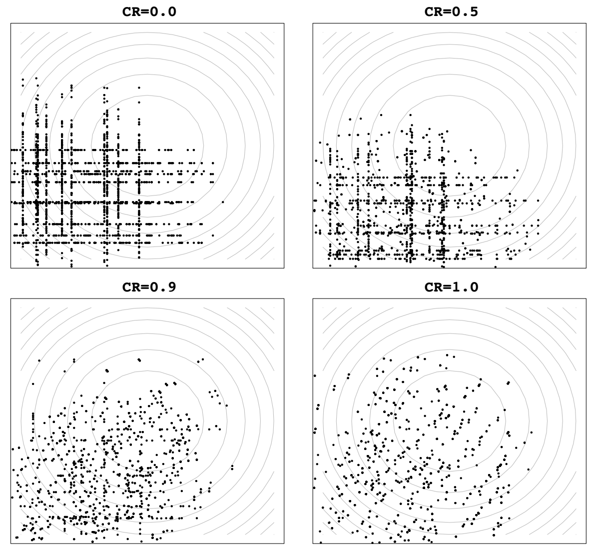

Figure 1: Empirical distributions of candidate trial vectors for $CR = 0$, $CR = 0.5$, $CR = 0.9$, and $CR = 1$. We ran DE on a single starting population of 10 members for 200 generations with replacement disabled to illustrate how trial vectors for this initial population were likely to lie.

Geometrically, changing more than one parameter at once results in a rotationally invariant search step. A few potential solutions for applying DE to non-separable functions have been given by employing either the current-to-rand [8, 4] or the rand/2/dir strategies [6]. However, we found that these strategies were not competitive with classic DE with a high CR on our selected benchmarks. This is likely because both variants are strict mutation-only strategies and crossover is still leveraging some artifact of the function. We found empirical support for this conjecture when we obtained similar results for a mutation-only rand/1/bin DE. Interestingly, this suggests that DE still must rely on recombination to some degree on the benchmarks we selected. Indeed, as Rönkkönen et al. [10], point out, occasional recombination can promote the algorithm's diversity and prevent stagnation.

2.1 Imposing Selective Pressure: Rank-based mutation

We can precisely define selective pressure as the ratio of probabilities between selecting the fittest individual and selecting the individual with median fitness. In classic DE's differential mutation, all vectors are equally likely to be selected as parents and donors. In this case, we say the selective pressure on being a parent or donor for differential mutation is 1.0: i.e. no selective pressure at all.

The differential mutation procedure allows DE to simulate a zero-mean anisotropic mutation distribution around the population. Price [8] argues that the distribution of mutant vectors created by sampling difference vectors models the ideal mutation distribution so long as the population can

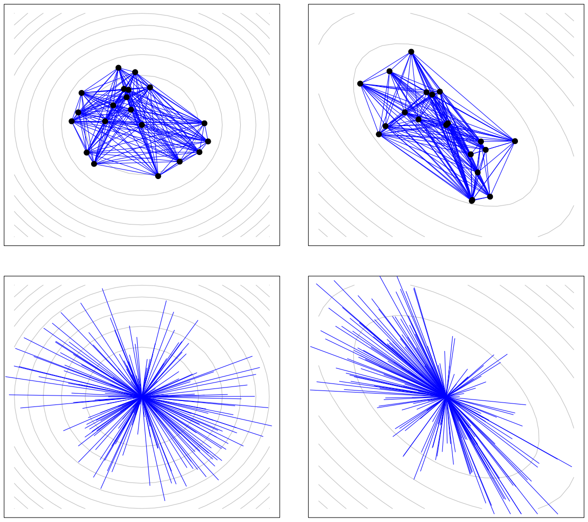

Figure 2: Differential vectors (top) and their mutation distribution (bottom) for elite points on a spherical (left) and elliptical (right) surface.

accurately sample the local topology of a function. Figure 2 illustrates this argument as a plot of vectors below a particular threshold and their centered difference vectors on two functions. The sub-threshold points on the spherical surface form a relatively spherical distribution and therefore produce fairly isotropic differential vectors. The sub-threshold vectors on the elliptical surface have a longer distribution in the direction of the semi-major axis, producing *longer* differential vectors along this direction, and thus larger mutation steps.

Mutation-only strategies have benefited from incorporating information about only the best individuals. For example, the rank-μ update in CMA-ES calculates the covariance of the μ best individuals and has been found to increase the efficiency of CMA-ES, especially for larger populations [7]. This update allows CMA-ES to learn distribution information from elite individuals in the population and can thus sample the local topology of the function better. In DE, the distribution information is "learned" by sampling random differential vectors from the population. Our conjecture is that restricting the set of these vectors to elite points in the population will have a similar effect on differential mutation, thus improving its efficiency.

In order to focus on elite differential vectors, we use a rank-based parent selection scheme to impose a bias on the selection step. This allows us to induce and control a selective pressure on the parent and donor vectors during differential mutation. Rank-based parent selection is obtained by sorting the population by fitness, and then drawing the parent and donor indexes (r_1, r_2, and r_3) from the linear distribution function given by Whitley [14]:

$$r_x := \left\lfloor \frac{\lambda}{2\,(\beta-1)} \cdot \left(\beta - \sqrt{\beta^2 - 4(\beta-1)U\,(0,1)}\right) \right\rfloor$$

where β is a bias term and $U(0,1)$ again denotes a random uniform distribution on $[0,1)$. The bias term controls the selective pressure. For example, if $\beta = 1.5$, then the index belonging to the top ranked individual is 1.5 times more likely to be selected than the individual with median rank. Note that we still impose the constraint $r_1 \neq r_2 \neq r_3 \neq i$.

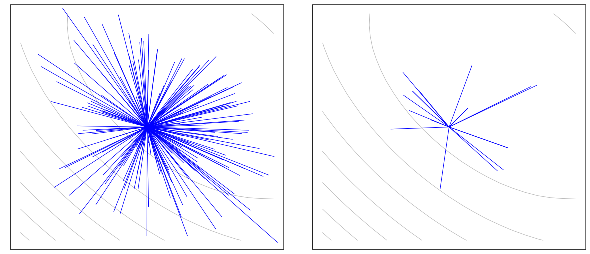

Figure 3: Differential vectors of a particular population that are selected uniformly (left) vs. differential vectors that were selected under a linear fitness-rank bias of $\beta = 3.0$ (right).

Figure 3 illustrates the effect by showing the differential vectors that are likely to be selected with and without selective pressure (left figure). The biased differential vectors (right figure) describe the function topology better and with fewer members. Similarly, the donor vector is more likely to be in a fitter region of the space.

3. EXPERIMENTS

In this Section we report results from a number of experiments designed to evaluate the hypotheses proposed in Section 1.

3.1 Reliance on crossover

Our first hypothesis was that, though DE relies heavily on crossover to produce axis-orthogonal steps on separable functions, it must rely more directly on differential mutation to solve non-separable functions. In order to test this hypothesis, we selected two functions and ran classic DE with two crossover parameters $CR = 0.1$ and $CR = 0.9$. Recall that $CR = 0.1$ results largely in axis orthogonal motion where $CR = 0.9$ is almost a mutation-only strategy (see Figure 1).

To control for modality we used a unimodal elliptic function and the multimodal Rastrigin function. The elliptic and Rastrigin are computed as follows

$$\begin{aligned} f_{elliptic}(x) &= \textstyle\sum_{i=1}^{n} 1000^{\frac{i-1}{n-1}} x_i^2 \\ f_{rast}(x) &= 10n + \textstyle\sum_{i=1}^{n} \left(x_i^2 - 10\cos(2\pi x_i)\right) \end{aligned}$$

where n is the function dimension. For this experiment we used $n = 30$. To control for separability, we ran the experiments using both non-rotated and rotated versions. To produce the rotated versions, we rotated each pair of parameters in a two dimensional subspace by a constant 5 degrees. For DE, we used $\lambda = 20$ and $F = 0.9$. We ran the algorithm for 100 trials on each variant and generated run length distributions. Each run length distribution is an empirical cumulative distribution function of the number of function evaluations the algorithm needs to find a solution within an ϵ-neighborhood of the global optimum. For our experiments we set $\epsilon = 10^{-5}$.

The run length distributions for these experiments are shown in Figure 4. In this figure we see an interesting effect between the two CR parameter settings. On the one

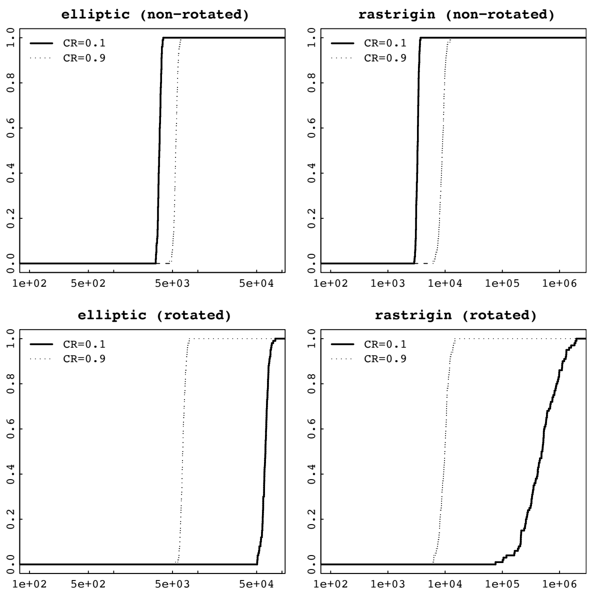

Figure 4: Empirical cumulative distribution of DE utilizing two different crossover parameters on non-rotated 5 dimensional functions (top) and rotated 5 dimensional functions (bottom). The x-axis reports the number of evaluations necessary to solve the function. $CR = 0.1$ (heavy line) produces a crossover motion with a rotationally dependent bias. $CR = 0.9$ (light line) is nearly a mutation-only strategy.

hand, when $CR = 0.1$ (heavy line), a low number of parameters are changed in each step resulting in axis-orthogonal motion. This motion is highly efficient on the separable functions. However, when the functions are rotated, there is a dramatic effect on the number of evaluations needed to solve the problem. On the other hand, the mostly-mutation setting $CR = 0.9$ (light line) appears largely impervious to the rotation.

Therefore, we have strong evidence that, when the function is non-separable, DE becomes more reliant on differential mutation, and less reliant on orthogonal steps produced by crossover. This leads us to our next hypothesis.

3.2 Effect of selective pressure

Our second hypothesis is that when DE must rely on differential mutation, selective pressure is inadequate to make efficient progress. To test this, we extend the work of Rönkkönen et al. [10] and their results with DE on the CEC 2005 special session on real parameter optimization. Specifically, we notice that DE has particular trouble with non-separable functions, an effect that is magnified as dimensionality increases. In their work, the authors note that higher population sizes will improve the convergence probability, but incur superfluous function evaluations. When performance becomes an issue (as it was in the contest) smaller population sizes are required, potentially at the expense of solution quality.

We therefore performed experiments to compare these previous results on a subset of benchmarks from CEC 2005 [13] with the rank-based variant of DE that imposes selective pressure. We first reproduced the results from Rönkkönen et al., and then applied selective pressure with three different biases, holding all other experiment parameters equal. We selected four benchmarks from the contest, controlling for *separability* and *modality*:

	separable	**non-separable**
unimodal	F_1 (sphere)	F_3 (rotated high condition elliptic)
multimodal	F_9 (Rastrigin)	F_{10} (rotated Rastrigin)

Note that all functions are shifted. The rotated versions were obtained using the orthogonal rotation matrices from the contest. Results are available from a set of 10 dimensional functions and a set of 30 dimensional functions. We chose to focus strictly on the 30 dimensional functions since DE seemed to struggle most with these.

We used the same parameters as Rönkkönen et al.: $F = 0.9$, $CR = 0.9$, $\lambda = 30$. We used three values for selective pressure, $\beta = 1.5$, $\beta = 2.0$, and $\beta = 3.0$. Following CEC 2005, we ran each algorithm variant on each function for 25 trials for 10^5 evaluations and found the median error convergence values. The results from our comparison are shown in Figure 5. The ranges for the CEC 2005 results are statistically identical to those reported by Rönkkönen et al. [10], except in the case of F_3 which was difficult to compare due to the high variance over only 25 trials.

We find a striking feature in Figure 5 comparing the convergences on F_9 and F_{10}. In particular, the latter is simply a *non-separable* version of the former. Classic DE seems to suffer significantly on the rotated version whereas there seems to be little effect on the rank-based variant with $\beta = 3.0$. The comparable median and minimum errors comparing the classic and rank-based variant in question are reported in Table 1. The rank-based variant makes slower progress than classic DE on the separable problem, but is not affected by the rotation. This supports our conjecture that selective pressure can provide a benefit on non-separable functions.

	F_9		F_{10}	
	median	*min*	*median*	*min*
classic DE	42.201	18.102	229.095	149.336
rank-based $\beta = 3$	60.692	35.818	65.667	36.813

Table 1: Relative median and minimum error of classic and rank-based DE variants on 30 dimensional F_9 and F_{10}. The rank-based variant, while slightly worse on the separable problem, seems to be impervious to rotation.

A similar effect seems to occur on F_3, but the reader is cautioned against misinterpreting the scale on the y-axis. The top performing algorithm (rank-based variant with $\beta = 3.0$), though significantly better than classic DE, still fails to find the solution (the best error value found is 3.22E+05). However, we believe this is an effect of the high condition (10^6) of the elliptic rather than its rotation. For this work, we are especially interested in the rotation transformation that provides non-separability.

To control for the condition effect, we ran the algorithms again on the same rotated elliptic with the condition tuned

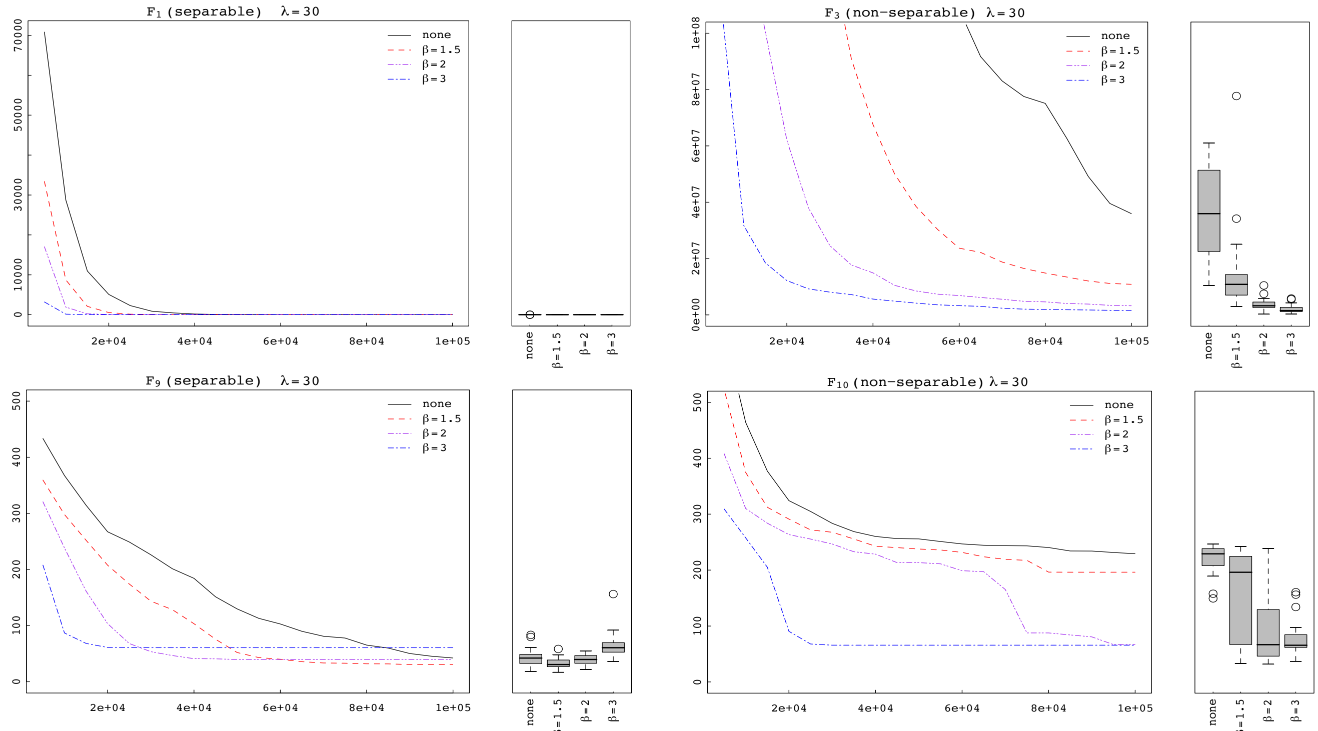

Figure 5: Median convergence error (y-axis) for DE on 30-dimensional benchmarks after 1E+05 evaluations (x-axis) with and without selective pressure (low population size). Results for unimodal functions are in the top row, results for multimodal functions are in the bottom row. Results for separable functions are in the left column, results for non-separable functions are in the right column.

down to 10^2. To control for separability, we used a non-rotated ($F_3^{lo,nr}$) and a rotated ($F_3^{lo,r}$) version. The results for the low-condition elliptics are displayed in Figure 6. All variants of DE were able to solve the separable low-condition elliptic. On the rotated low-condition elliptic, only the rank-based variant with $\beta = 3.0$ solved the function in all 25 trials where classic DE with no selective pressure did not solve it at all during the allotted time. This lends further support to our conjecture that selective pressure focuses differential mutation, and is therefore beneficial on non-separable problems.

3.2.1 *Exploitative power*

We have evidence that imposing selective pressure is providing an advantage over the classic DE variant, especially on the non-separable functions. A natural follow-up question is thus: by what mechanism does selective pressure gain this advantage? We conjecture that it helps to *focus* the search; that is, it puts a needed emphasis on exploitation. This may be beneficial when the algorithm is depending more on differential mutation, as we would expect on non-separable functions.

To test the "focusing power" of the search variants, we observe change in population diversity over time. To measure this, the *population dispersion* δ is computed as

$$\delta = \frac{1}{\binom{\lambda}{2}} \sum_{i=1}^{\lambda-1} \sum_{j=(i+1)}^{\lambda} \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T (\mathbf{x}_i - \mathbf{x}_j)}$$

that is, the average pair-wise Euclidean distance between all population members. To measure the relative population dispersions on the functions, we ran the $\lambda = 30$ variants once again for 25 trials and measured the mean population dispersion. The results from this experiment are plotted (on a log scale) in Figure 7.

To see when the difference in dispersion becomes statistically significant, we performed a Wilcoxon signed rank test on the difference between each rank-based variant and the classic DE variant. We present the p-values over (log) time in Figure 8. From these results we can see that the difference in dispersion between the classic variant and each of the rank-based variants becomes statistically significant on all functions by at most 500 evaluations.

Unsurprisingly, it is clear that the rank-based variants are creating a lower population dispersion which benefits differential mutation by focusing the search. This raises some important questions regarding stagnation, however, which will be addressed in Section 4.

3.2.2 *Increasing the population size*

The results from the foregoing experiments suggest that selective pressure adds an advantage to DE's progress by focusing the search. Rönkkönen et al. point out that one of the issues with DE's non-convergence was that λ should be between 200 and 600 to optimize difficult functions [10]. However, one issue with large population sizes is speed. We conjecture that selective pressure might offer a distinct speed-up with high population sizes since it appears to focus the mutation process better. To test this, we ran the variants on the same (30 dimensional) functions with pop-

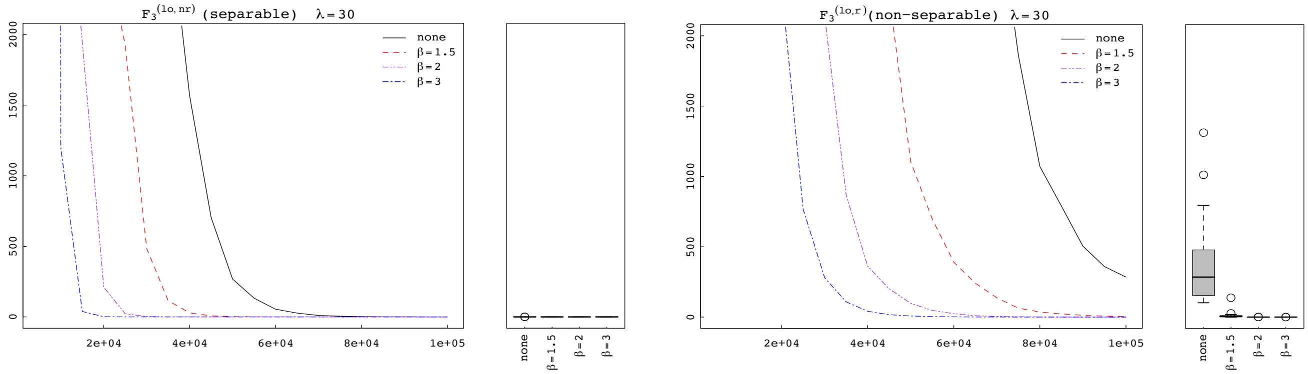

Figure 6: Controlling for condition: median convergence error (y-axis) for DE after 1E+05 evaluations (x-axis) with low population size on $F_3^{lo,nr}$ (low-condition non-rotated elliptic), and $F_3^{lo,r}$ (low-condition rotated elliptic).

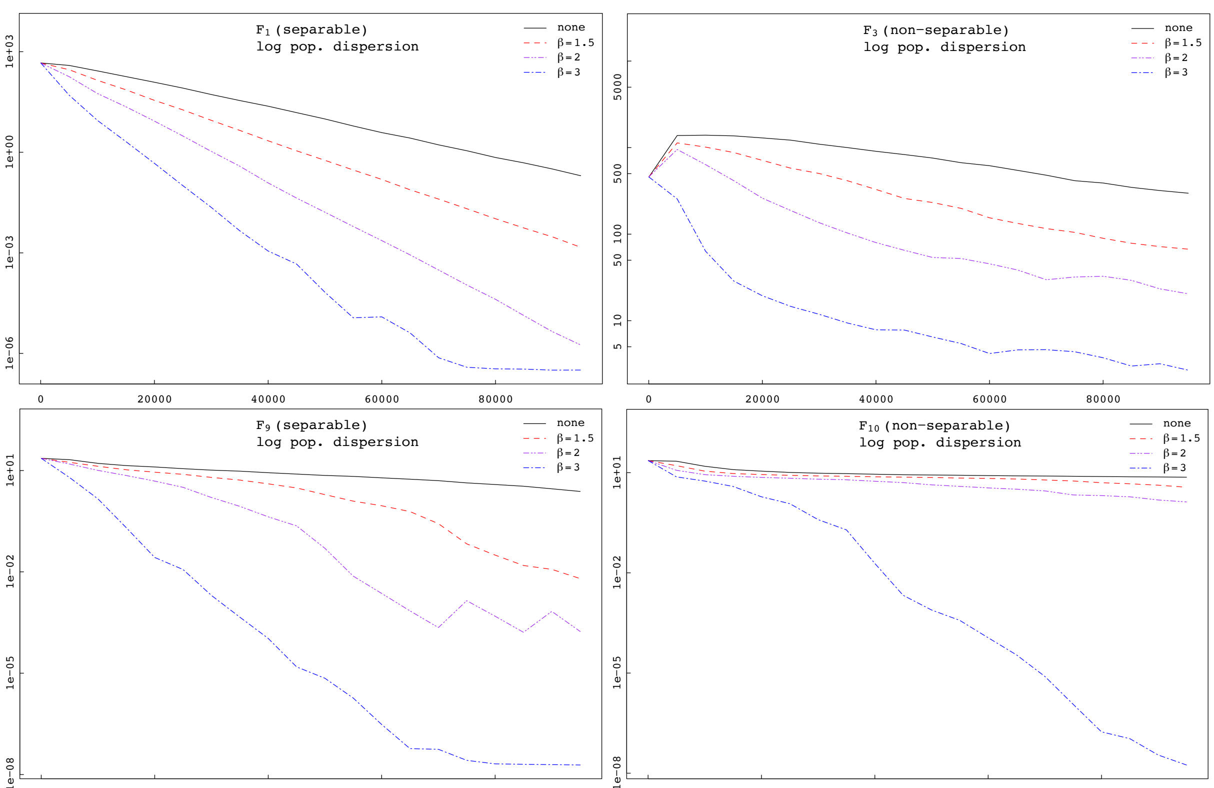

Figure 7: Log population dispersion (y-axis) of DE variants on each function for 1E+05 (x-axis) evaluations. Population size is $\lambda = 30$.

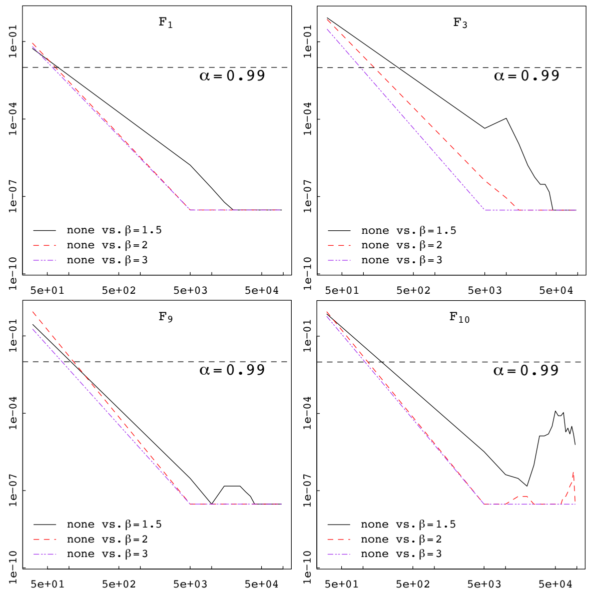

Figure 8: Results from Wilcoxon signed rank tests. Log p-value vs. log evaluations. Each plotted line represents the p-value of a statistical test that a rank-based variant has significantly lower dispersion than the classic DE variant at a particular point during search. Lines below the $\alpha = 0.99$ line are significant at 99% confidence.

ulation size $\lambda = 500$ for 10^7 evaluations. Our goal was to observe the relative convergence speeds.

The results from the high population size experiments are shown in Figure 9. It is clear that the population size increase is causing a dramatic reduction in convergence speed. Evidence that selective pressure gives an advantage when the population size is high comes in the form of faster (and further) median progress on the rank-based variants. It is important, however, to note that except in the case of F_1, none of the high population variants succeed in solving the function, even after ten million evaluations.

4. DISCUSSION

We hypothesized that 1) DE relies more on differential mutation than rotation dependent crossover on non-separable functions and 2) DE's weak selective pressure may result in inefficient exploitation when relying on differential mutation. Our results support both these hypotheses. We found that the high mutation strategy, though slightly less efficient on separable problems, is invariant to rotation. This can be contrasted with the axis orthogonal motion incurred by changing a few parameters at a time, as with low CR values.

Our results also indicate that increasing selective pressure on the vectors selected as parents and donors for differential mutation increases the exploitative power of the search by focusing the diversity of differential mutation. This effect is seen dramatically on non-separable functions when differential mutation becomes important.

However, the algorithm still performs uncompetitively on non-separable, multimodal functions. In fact, none of the variants were able to solve either F_9 or F_{10} in any of our experiments. One surprising result was that increasing the population size still did not cause convergence as predicted by Rönkkönen et al. [10] even after ten million function evaluations.

Of course, we cannot rule out the possibility of poor parameter selection. We wanted to remain as faithful as possible to the results reported for the CEC 2005 competition [10]. This restricted our parameter settings. However, pilot experiments with different settings did not improve the results by a great deal. We hypothesize that, as function dimension increases, the differential vectors in a population do not provide enough information to accurately estimate the ideal distribution.

One inherent danger with high selective pressure is the rapid loss of diversity within the population. The increase of exploitative power comes at a cost to potential exploration. On some trials we see the characteristic stagnation effect from the lack of diversity. This highlights the importance of a good balance of exploration and exploitation in search. Perhaps on more difficult functions, DE would benefit from a time-variable selective pressure. Diversity might also be injected by using a soft restart mechanism [5].

The rank-based variant, unlike the classic DE algorithm, calls for the population to be sorted in each generation. This requirement can be removed by using tournament selection which produces equivalent behavior to linear rank-based selection in expectation [2] for $\beta \leq 2.0$. When $\beta > 2.0$, the selective pressure engendered by the tournament method becomes mildly non-linear.

5. CONCLUSION

Differential Evolution is a popular parameter optimization algorithm in the Evolutionary Computation community. We have shown that its failure to produce competitive results on non-separable functions may be attributable in part to inadequate selective pressure when choosing parent and donor vectors for differential mutation. On non-separable functions, when the algorithm must rely heavily on this mutation, this lack of exploitative power causes significantly slower progress.

Furthermore, selective pressure can also offer an advantage when the population size is high, presumably by preferring a smaller number of elite individuals for direction and step size information. We see faster convergence with the rank-based variants, even on separable problems.

One direction of future work is to introduce an annealing schedule to the β term that allows early exploration, but promotes intensification later in the search. This could help mitigate potential stagnation problems.

A lack of competition arises from DE's parallel selection strategy. The generational selective pressure between parents and offspring might also be increased by using a different selection strategy. Another avenue of future work is to experiment with such selection techniques.

6. REFERENCES

[1] Thomas Bäck. Selective pressure in evolutionary algorithms: a characterization of selection mechanisms. In *Proceedings of the First IEEE Conference on Evolutionary Computation*, 1994.

[2] David E. Goldberg and Kalyanmoy Deb. A comparative analysis of selection schemes used in

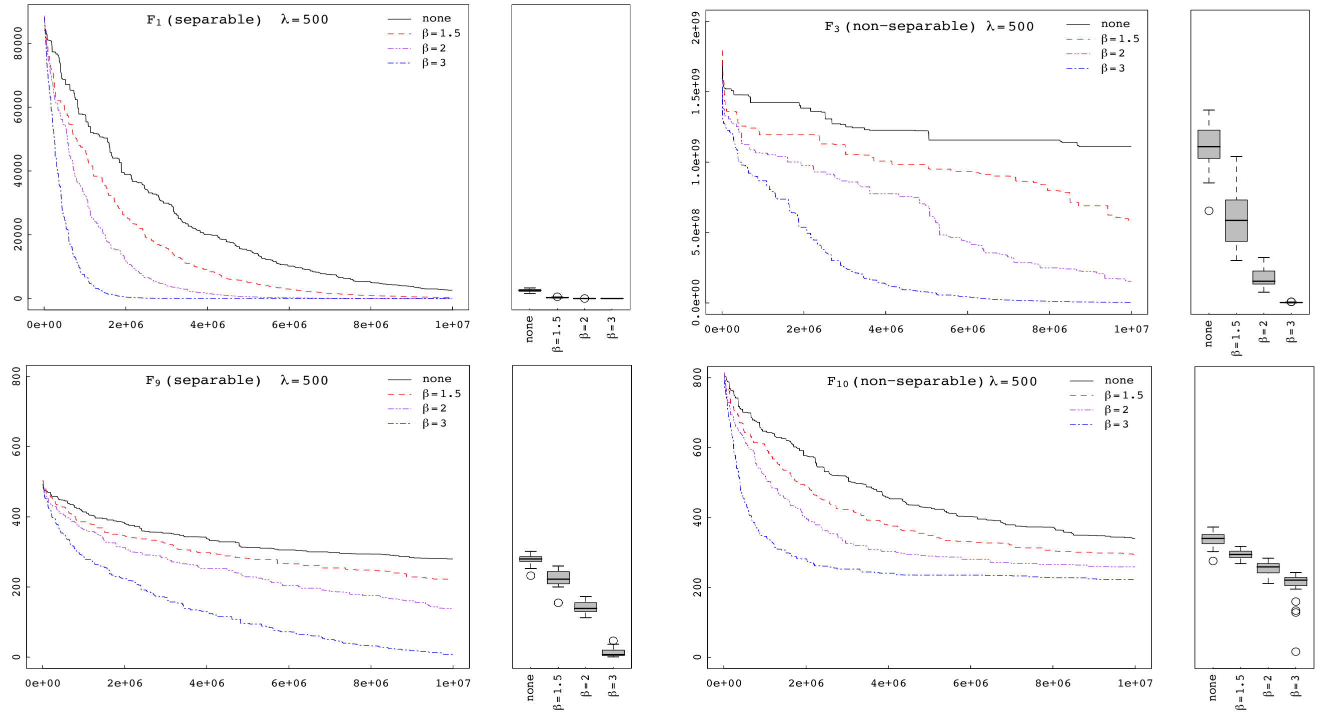

Figure 9: Median convergence error (y-axis) for DE on 30-dimensional benchmarks after 1E+07 evaluations (x-axis) with and without selective pressure (High population size). Results for unimodal functions are in the top row, results for multimodal functions are in the bottom row. Results for separable functions are in the left column, results for non-separable functions are in the right column.

genetic algorithms. In Gregory J. E. Rawlins, editor, *Foundations of Genetic Algorithms*, pages 69–93. Morgan Kaufmann, 1991.

[3] Nikolaus Hansen and Stefan Kern. Evaluating the CMA evolution strategy on multimodal test functions. In *Proceedings of 8th International Conference on Parallel Problem Solving from Nature*, pages 282–291, Berlin, Germany, 2004.

[4] Antony W. Iorio and Xiaodong Li. Solving rotated multi-objective optimization problems using differential evolution. In *Proceedings of the 17th Joint Australian Conference on Artificial Intelligence*, 2004.

[5] Keith E. Mathias, J. David Schaffer, Larry J. Eshelman, and M. Mani. The effects of control parameters and restarts on search stagnation in evolutionary programming. In *PPSN V: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, pages 398–407, London, UK, 1998. Springer-Verlag.

[6] Efrén Mezura-Montes, Jesús Velásquez-Reyes, and Carlos A. Coello Coello. A comparative study of differential evolution variants for global optimization. In *Genetic and Evolutionary Computation Conference (GECCO 2006)*, 2006.

[7] S.D. Müller, N. Hansen, and P. Koumoutsakos. Increasing the serial and the parallel performance of the CMA-Evolution Strategy with large populations. In *Parallel Problem Solving from Nature (PPSN 02)*, 2002.

[8] Kenneth V. Price. An introduction to differential evolution. In *New ideas in optimization*, pages 79–108. McGraw-Hill Ltd., UK, Maidenhead, UK, England, 1999.

[9] Kenneth V. Price and Rainer Storn. Differential evolution. *Dr. Dobb's Journal*, 264:18–24, 1997.

[10] Jani Rönkkönen, Saku Kukkonen, and Kenneth V. Price. Real-parameter optimization with differential evolution. In *The 2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 506– 513, Sept 2005.

[11] Ralf Salomon. Raising theoretical questions about the utility of genetic algorithms. In Peter J. Angeline, Robert G. Reynolds, John R. McDonnell, and Russ Eberhart, editors, *Evolutionary Programming VI*, pages 275–284, Berlin, 1997. Springer.

[12] Rainer Storn and Kenneth V. Price. Differential Evolution: a fast and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.

[13] P.N. Suganthan, N. Hansen, J.J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari. Problem definitions and evaluation criteria for the CEC 2005 special session on real parameter optimization. Technical report, Nanyang Technological University, Singapore, 2005.

[14] L. Darrell Whitley. The GENITOR algorithm and selection pressure: why rank-based allocation of reproductive trials is best. In *Third International Conference on Genetic Algorithms*, San Mateo, CA, 1989. Morgan Kaufman.