Control and Cybernetics

vol. **26** (1997) No. 3

Diagonal crossover in genetic algorithms for numerical optimization

by

A.E. Eiben* and Cees H.M. van Kemenade**

* Leiden University,
e-mail: gusz@wi.leidenuniv.nl

** CWI, Amsterdam,
e-mail: kemenade@cwi.nl

Abstract: In this paper the results of a detailed investigation on a multi-parent recombination operator, diagonal crossover, are reported. Although earlier publications have indicated the high performance of diagonal crossover on a number of problems, so far it has not been investigated whether high performance is indeed a result of using a high number of parents. Here we formulate three hypotheses to explain why GA performance increases when more parents are used. Based on an extensive study on a test suite containing eight numerical optimization problems we are able to establish that the higher number of parents is indeed one of the sources of higher performance, if and when this occurs. By the diversity of the test functions (unimodal, multimodal, quasi-random landscapes) we can also make observations on the relationship between fitness landscapes and operator performance.

Keywords: genetic algorithms, numerical optimization, recombination, multi-parent crossovers

1. Introduction

Multi-parent recombination is a new research area within evolutionary computation. Although some researchers have incidentally proposed and applied recombination mechanisms using more parents, Bersini and Seront (1996), Bremermann, Rogson and Salaff (1966), Mühlenbein (1989), the *phenomenon* of multi-parent recombination has not been given much attention in the past. In this paper we study this phenomenon by investigating the behavior of diagonal crossover (see the definition in Section 2). Our research goals are two-fold.

1. We try to find connections between the structure of the fitness landscape and the performance of diagonal crossover. In particular we want to establish on what kind of landscapes it is advantageous to increase the number of parents. (Diagonal crossover for 2 parents is identical to the traditional 1-point crossover operator.)
2. We try to disclose the source of increased performance of the diagonal crossover with more parents when it is superior to 2-parent recombination.

Further elaboration of the second research objective requires technical details, and so, we return to this issue after the exact definition of the operator. The rest of the paper is organized as follows. In the following section we briefly review multi-parent recombination operators in Evolutionary Algorithms. After the exact definition of diagonal crossover we formulate three hypotheses that can clarify why the performance of the GA increases when the number of parents in diagonal crossover is increased. To this end we design experiments that allow rejection or confirmation of these hypotheses. In Section 4 we present the GA used in the experiments and discuss the performance measures to be used to monitor GA performance. The test suite and the results of the experiments on each test function are presented in Section 4. Finally, in Section 5 we evaluate the results and draw conclusions.

2. Multi-parent recombination

In evolution strategies (ES) global recombination is a multi-parent operator, Bäck (1996), Schwefel (1995). This operator creates a new value in the child chromosome based on two parents, but randomly chooses two parents for each variable anew. By this particular mechanism the number of parents is undefined, thus investigations on the effects of different number of recombinants on algorithm performance could not be performed in the traditional ES framework. (Let us note that in Schwefel and Rudolph, 1995, an extension of ES is proposed that allows tuning of the number of recombinants.) So far there are almost no experimental results available on the (dis)advantages of global recombination with respect to usual, two-parent recombination. Schwefel (1995) briefly touches on this issue stating that 'appreciable acceleration' is obtained by changing to bisexual from asexual scheme (i.e. adding recombination using two parents to the mutation-only algorithm), but only 'slight further increase' is obtained when changing from 'bisexual to multisexual recombination' (i.e. using global recombination instead of the two-parent variant).

Related work of Beyer (1995), generalizes the traditional ES recombination operators by introducing the number of parents as an independent parameter ρ. The resulting $(\mu/\rho, \lambda)$ evolution strategy is studied for the special case of $\rho = \mu$ and theoretical analysis on the spherical function shows an advantage of using more than two parents.

Global recombination in ES also fertilized Genetic Algorithms. The gene-pool recombination of Mühlenbein and Voigt mixes information of possibly more parents by a similar mechanism as global recombination in ES, Mühlenbein and Voigt (1995), Voigt and Mühlenbein (1995). Hence, the number of parents is not defined here either. Mühlenbein and Voigt report an increase of performance when using gene-pool recombination (GPR) instead of two-parent recombination (TPR). GPR is showed to be approximately 25% faster than TPR on the ONEMAX problem, and the fuzzyfied GPR outperforms TPR on the spherical function in speed and in realized heritability.

The N-parent generalizations of the traditional 1-point crossover and uniform crossover in GAs were introduced in Eiben, Raué and Ruttkay (1994). The resulting diagonal, respectively, scanning crossover have the number of parents as parameter, and therefore are tunable on the 'extent of sexuality'. This tunability is new feature compared to global recombination and gene pool recombination, where the multi-parent option can only be switched on or off, but it is not scalable. Several studies, e.g. Eiben, van Kemenade and Kok (1995), Eiben, Raué and Ruttkay (1994), Eiben and Schippers (1996), van Kemenade, Kok and Eiben (1995), have shown that using more than two parents in either crossover mechanism can increase GA performance, although this does not hold for every problem and the two operators can respond differently to increasing the number of parents.

The main subject of the present investigation, diagonal crossover, generalizes 1-point crossover for N parents by selecting $(N-1)$ crossover points and composing N children by taking the resulting N chromosome segments from the parents 'along the diagonals'. The idea is illustrated for $N = 3$ in Fig. 1 up.

With respect to our second research objective let us make the following observations. First, the increase in the number of parents automatically leads to an increased number of crossover points. It can be the case that higher performance for higher N's is not the result of using more parents, but simply comes from being more disruptive by using more crossover points. This forms our first working hypothesis.

H_1 Using more crossover points leads to better performance.

Second, notice that by application of the diagonal crossover, N parents create N children in one go. Since we use a steady state GA and update the population, i.e. insert offspring, after each application of crossover (followed by mutation), this means that a GA using 10 parents diagonal crossover has more information before performing the selection step than a GA using the two-parents version. In other words, GAs with higher operator "arity" have a bigger generational gap which might cause a bias in their favor. Our second working hypothesis is accordingly the following.

H_2 Bigger generational gap leads to better performance.

Finally, we maintain our original conjecture that the advantages of using diagonal crossover with higher "arities" are not the result of an unintended artifact.

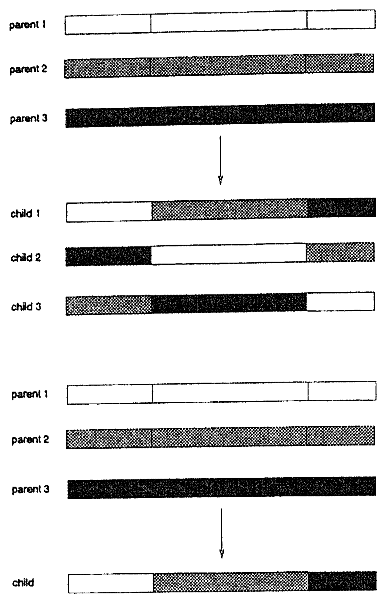

Figure 1. Diagonal crossover with three parents and three children (up) and with three parents and one child (down).

H_3 Using more parents leads to better performance.

Note that the hypotheses H_1, H_2 and H_3 are not mutually exclusive as there might be more sources of increased GA performance when increasing N in diagonal crossover. The main contribution of the present paper is that these sources are investigated in isolation hence providing a solid ground to check whether higher performance for higher N's is an artifact (H_1, H_2), or the higher number of parents is indeed advantageous.

3. Experiment setup and performance measures

All experiments are executed using a GA setup as described in Table 1. A non-standard option is the uniform random parent selection mechanism, whereby no selective pressure is applied when choosing recombinants. The motivation comes from van Kemenade, Kok and Eiben (1995), where we observed that this mechanism is preferable. Note that uniform random parent selection mechanism is standard in Evolution Strategies.

In order to test the working hypotheses presented in Section 2 we run experiments with a set of different crossover operators. For investigating H_1 we apply the traditional two-parent two-children N-point crossover, De Jong and Spears (1992). This operator is well known from the literature, therefore we omit a def-

representation	fixed point binary with Gray coding
GA type	steady state
parent selection	uniform random
deletion mechanism	worst fitness deletion
number of parents	2-30
crossover rate p_c	1.0
mutation rate p_m	1/chromosome length
population size	500
termination criterion	population converged OR optimum hit
max. nr. of evaluations	100,000
results averaged over	50 independent runs

Table 1. GA setup used in the experiments

inition. If N-point crossover exhibits increasing performance when increasing N, (the experimental results reported in Eiben, van Kemenade and Kok, 1995, make us expect this) then we accept H_1. To test the second hypothesis H_2, we will apply a slightly modified version of diagonal crossover that creates only one child. The lower part of Fig. 1 illustrates this operator. When we use the one child version of diagonal crossover the generational gap does not increase with increasing the number of parents. If the original variant outperforms the one child version of diagonal crossover, then we accept the hypothesis H_2. Concerning hypothesis H_3, note that the number of chromosome segments using N-point crossover is $N + 1$, which equals the number of chromosome segments obtained by diagonal crossover for $N + 1$ parents. This means that the disruptiveness of these operators grows parallelly as N increases. If higher disruptiveness increases GA performance on our test suite, then the performance of both the N-point crossover and the diagonal crossover will increase with increasing N. This, however, does not imply that more parents have no additional advantage as the performance of diagonal crossover might grow faster with increasing N than that of the two-parent N-point crossover. We accept the hypothesis H_3 if diagonal crossover for $N + 1$ parents is better than N-point crossover.

To evaluate different GA setups, that is, the effect of different number of parents, respectively crossover-points, several performance measures of a run are monitored. The two main performance measures are accuracy and speed. Accuracy is measured by the error at termination. Since all functions have a minimum of zero, we use the best objective function value at termination as the accuracy measure of one run, and the median of the best objective function values, calculated over the 50 independent runs, as the accuracy belonging to a specific setting. For practical purposes we consider 10^{-10} as zero and terminate the run if this value is achieved. Let us remark that using medians instead of average values has an advantage, namely medians are less sensitive for outliers in the data. On the other hand, if the optimum is found in the majority of the runs, then the median will equal the optimum. Additionally to the medians of the outcomes we also present the 99% confidence interval bars to the performance curves. The second main performance measure is the speed of the algorithm, measured by the median number of fitness evaluations before termination. If the GA with a certain setup never finds the optimum, this value equals the maximum number of fitness evaluations. A third performance measure is the success rate, i.e. the percentage of runs where the optimal objective function value has been found. We will present figures on success rates and 90% confidence intervals, whenever the accuracy or the speed curves are (nearly) constant, thus providing (almost) no basis to compare different settings. Finally, for a detailed insight in the behavior of the GA sometimes we also depict the progress curves of the evolution for 18 parents (diagonal crossover), respectively 17 crossover points (N-point crossover). These curves (with a logarithmic y-axis) show the population's best objective function value as a function of the number of executed fitness evaluations, averaged over 50 independent runs.

4. Experimental results

The experiments have been performed on eight numerical function optimization problems. Each function is to be minimized and is scaled to have an optimal function value of zero. The fitness landscapes defined by these functions have various characteristics, unimodal, multimodal and quasi-random, i.e. very rugged with randomly distributed local optima. Additionally, some of the functions are separable, while others are not. The exact definitions will be given in the corresponding subsections, here we only give a summary on their separability, the dimensions and the representation used in the experiments. As default, we use 20 bits for representing a single variable, but deviate from this value for F1, F2 and F8. For F1 and F2 we use the values originating from de Jong, for F8 30 bits are used, following Bäck (1996). A concise treatment on numerical optimization problems as test functions can be found in Bäck and Michalewicz (1997).

Property	F1	F2	F3	F4	F5	F6	F7	F8
separable	y	n	n	n	y	y	y	n
dimension	3	2	30	10	10	10	10	30
chrom. length	30	22	600	200	200	200	200	900

Table 2. Properties of the test functions

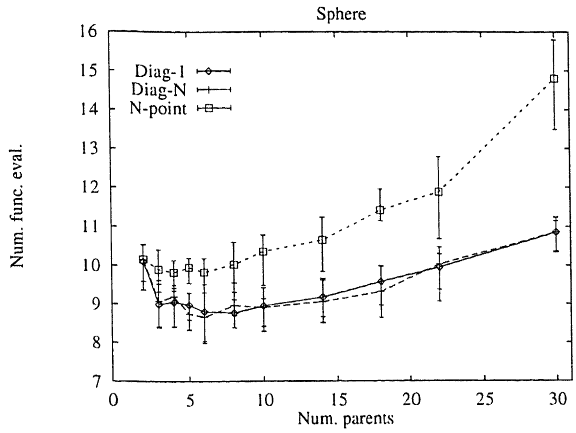

Figure 2. Speed curves for the spherical function

4.1. Spherical function

The first test function is the spherical function:

$$F1(\vec{x}) = \sum_{i=1}^{n} x_i^2 ,$$

where $-5.12 \leq x_i \leq 5.12$. This function is one of the most widely used objective functions in Evolutionary Computation, especially for convergence velocity evaluation. It has a unimodal, smooth fitness surface and is separable, making optimization rather easy. We tested the classical version of de Jong with $n = 3$. The GA found the optimum with every setting (every operator, for every value of N). Therefore we omit accuracy and success rate data, only presenting the speed curves in Fig. 2.

From the speed curves it turns out that the two variants of diagonal crossover show almost identical behavior and both are faster than N-point crossover. Furthermore, it seems that there is a limit to increasing N: approximately up to 6 it leads to performance increase, thereafter the performance begins to deteriorate.

4.2. Rosenbrock's saddle function

F2 is the saddle function after Rosenbrock:

$$F2(\vec{x}) = 100 \cdot (x_1^2 - x_2)^2 + (1 - x_1)^2,$$

where $-2.048 \leq x_i \leq 2.048$. The global minimum is zero at $\vec{x} = (1, 1)$. The Rosenbrock function is not separable and the unimodal fitness landscape is characterized by an extremely deep valley along the parabola $x_1^2 = x_2$.

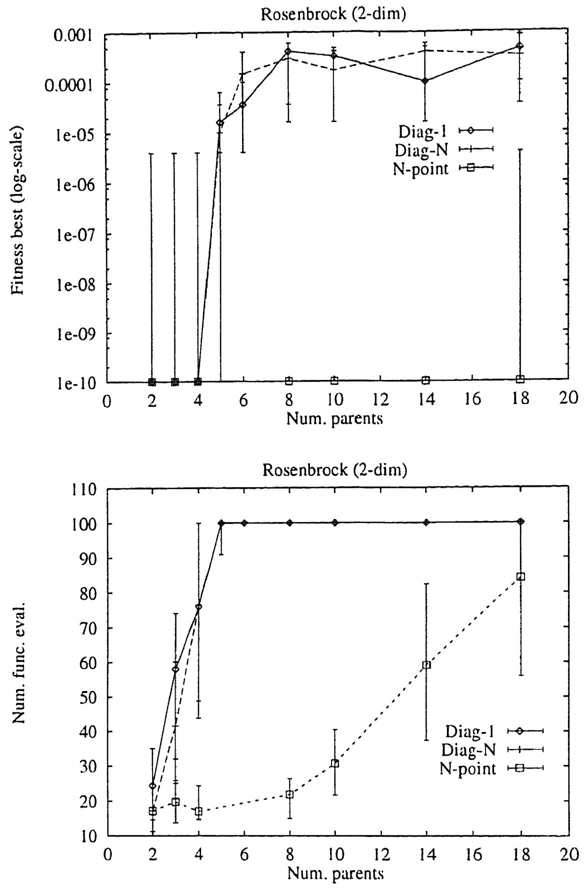

Figure 3. Accuracy (upper diagram) and speed (lower diagram) for the Rosenbrock function

Recall from Table 2 that we use the classical de Jong setting with chromosome length 22 for F2. Therefore, the maximum number of parents is lowered accordingly in these experiments.

The accuracy and speed curves suggest that increasing N decreases the performance. The success rate curves in Fig. 3 disclose that this is only partly true. The optimization performance grows with N for N-point crossover (up to $N = 8$), but deteriorates for the diagonal crossovers. N-point crossover outperforms both diagonal crossovers with respect to each performance measure. The two variants of diagonal crossover are practically identical for F2.

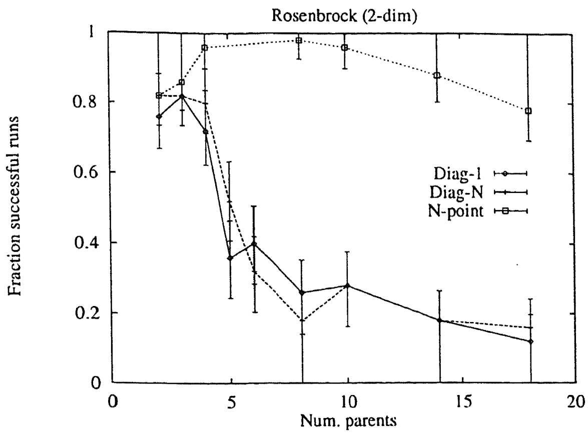

Figure 4. Success rates for the Rosenbrock function

4.3. Ackley function

Our third test function F3 is the Ackley function:

$$F3(\vec{x}) = 20 + e - 20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right),$$

where $-30 \leq x_i \leq 30$. The global minimum of zero is at $\vec{x} = (0,0,0,\ldots)$. This function is not separable and at a low resolution the fitness landscape is unimodal, but the second exponential term covers the landscape with many small peaks and valleys, i.e. many local optima. We tested F3 for $n = 30$ and observed that the GA never found the optimum. Accordingly, the speed and the success rate curves are constant, therefore omitted here. We present the accuracy curves in Fig. 5.

The effect of higher N's is clear from the accuracy curves. Increasing N is advantageous for each operator up to the upper limit we tested. The one child and the N-children versions of diagonal crossover perform identically also on this function, and both diagonal crossovers are consistently better than N-point crossover.

4.4. Griewangk function

F4, the Griewangk function is defined as follows.

$$F4(\vec{x}) = 1 + \sum_{i=1}^{n}\frac{x_i^2}{400n} - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right),$$

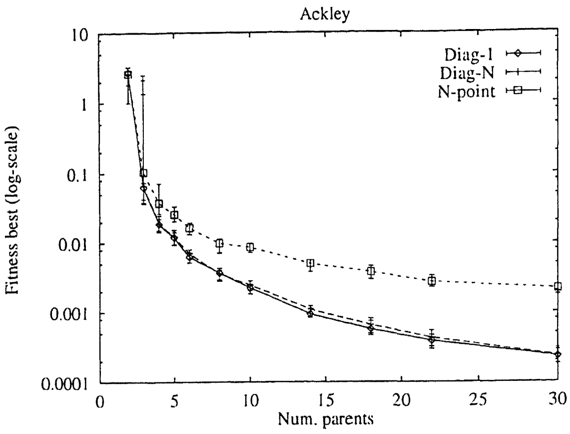

Figure 5. Accuracy curves for the Ackley function

where $-600 \leq x_i \leq 600$. The global minimum of zero is at the point $\vec{x} = (0, 0, 0, \ldots)$. This function has a product term introducing an interdependency between the variables, thus it is not separable. F4 was tested for 10 dimensions, the results are exhibited in Fig. 6.

On this function the advantages of higher N's are clear, but the performance increase of accuracy stops at about $N = 15$. While the accuracy curves show only modest differences between the operators, the results on the speed of the algorithm disclose that the diagonal crossovers are significantly faster after $N = 5$. The two variants of diagonal crossover do not differ significantly.

4.5. Michalewicz function

The fifth test function is taken from Michalewicz, Bersini, Dorigo, Langerman, Seront and Gambardella (1996).

$$F5(\vec{x}) = -\sum_{i=1}^{n} \sin(x_i) \cdot \sin^{2n}\left(\frac{i x_i^2}{\pi}\right),$$

where $0 \leq x_i \leq \pi$. We tested F5 for $n = 10$ and observed that the GA found the optimum in the majority of runs. Hence the medians of the accuracy results are equal to the optimal value. Therefore we rather present the success rates instead of the accuracy data.

Increasing N above 2 on the Michalewicz function results in the highest gains so far. The success rates show a spectacular increase from approximately 30% for 2 parents to 80-90% for N between 5-20 and the GA becomes approximately four times faster for $N = 5 - 10$, than for $N = 2$. Comparing the operators we see again the superiority of diagonal crossover and no significant difference between the one child and the N-child version.

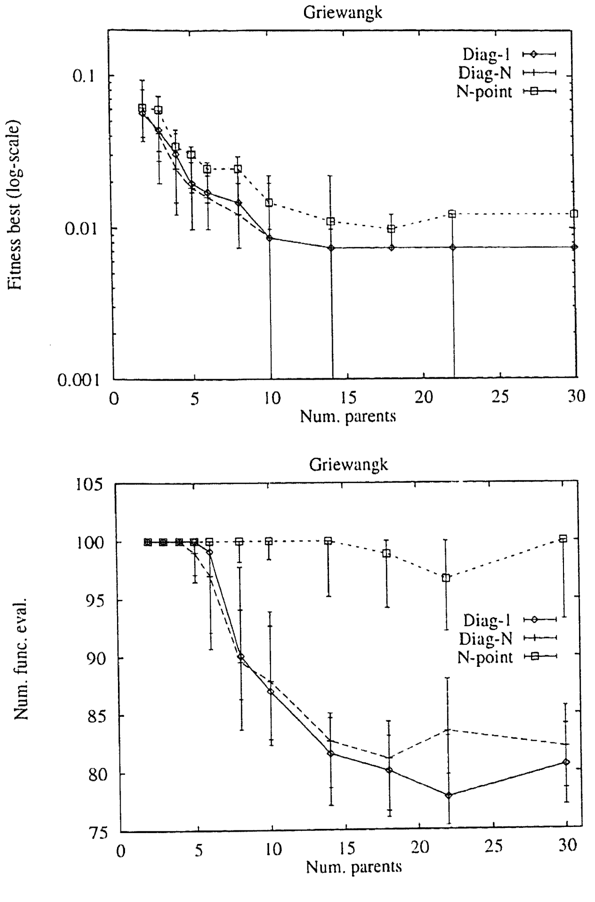

Figure 6. Accuracy (upper diagram) and speed (lower diagram) for the Griewangk function

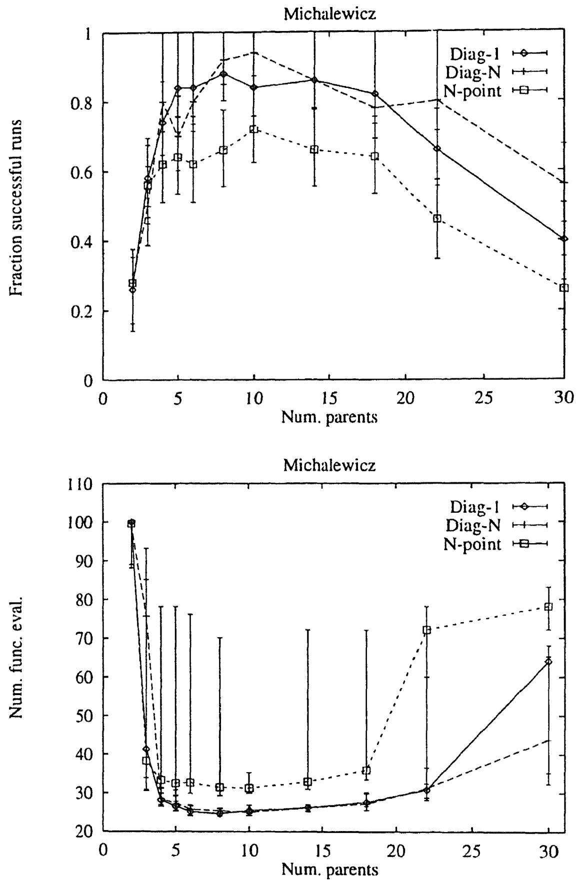

Figure 7. Success rates (upper diagram) and speed (lower diagram) for the Michalewicz function

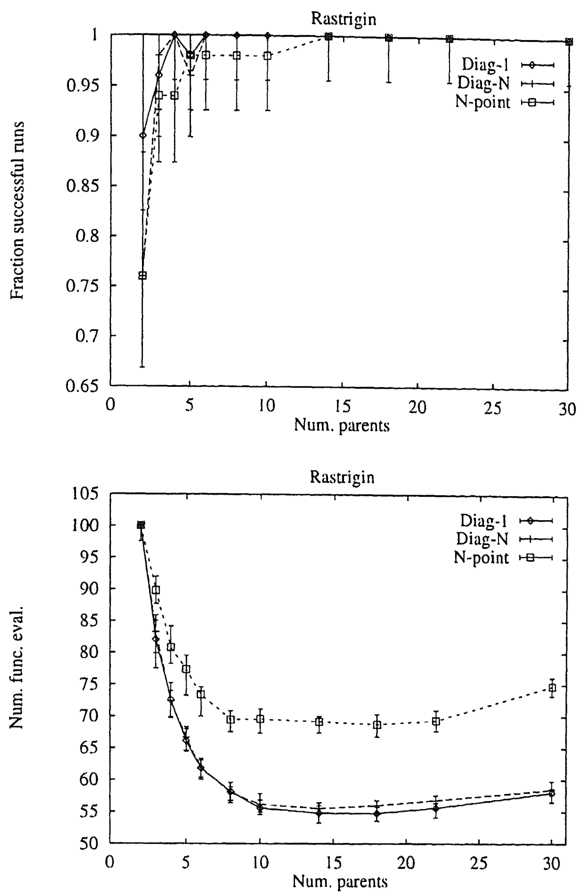

Figure 8. Success rates (upper diagram) and speed (lower diagram) for the Rastrigin function

4.6. Rastrigin function

F6 is the Rastrigin function:

$$F6(\vec{x}) = \alpha n + \sum_{i=1}^{n} x_i^2 - \alpha \cos(2\pi x_i),$$

where $-5.12 \leq x_i \leq 5.12$. The global minimum of zero is at $\vec{x} = (0, 0, \ldots)$. This function is separable and its primary characteristic is the existence of many suboptimal peaks whose values increase as the distance from the global optimum point increases. In our tests we used $\alpha = 10.0$ and $n = 10$.

Since many runs found the optimum, accuracy figures are replaced by success rates curves (notice the 0.65 - 1.0 scaling in Fig. 8, upper part). These show that

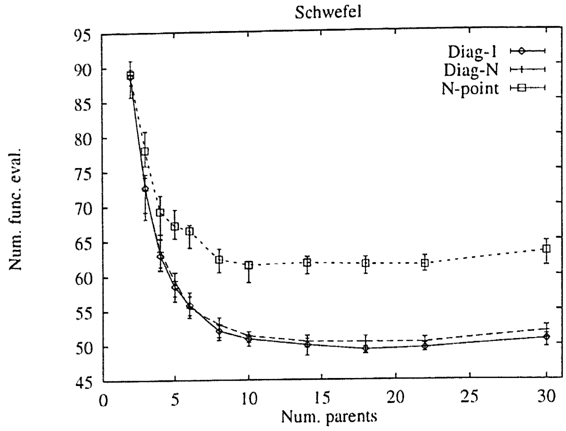

Figure 9. Speed curves for the Schwefel function

increasing N is advantageous, but the differences between various operators are small. Looking at the speed curves the effect of higher N's and the differences between operators become clear. We can observe that each operator becomes better for higher N's and that the two diagonal crossovers (identical again) outperform N-point crossover.

4.7. Schwefel function

F7 is obtained by generalizing Schwefel's 2.26 function (Schwefel, 1995, p. 344):

$$F7(\vec{x}) = 418.9829n - \sum_{i=1}^{n} x_i \sin\left(\sqrt{|x_i|}\right),$$

where $-512.03 \leq x_i \leq 511.97$. The global minimum of zero is at $\vec{x} = (420.9687, 420.9687, \ldots)$. Although this function is separable, it is interesting because of the presence of a second-best minimum far away from (in the 'opposite corner' to) the global minimum. This feature, just like two-peaks landscapes, makes the GA sensitive for early commitment with respect to the search direction. F7 was tested for $n = 10$ and turned out to be easy. Nearly all runs ended with the global optimum, implying that accuracy and success rates would give no information for comparing the operators. The results on speed, however, show that the GA performance quickly and consequently improves when increasing N from 2 to approximately 10-15, and stagnates thereafter. The algorithm becomes approximately twice as fast for high N's as for $N = 2$. Once again, there is no significant difference between the two diagonal crossovers that both outperform N-point crossover.

4.8. Fletcher-Powell function

F8, the Fletcher-Powell function is retrieved from Bäck (1996):

$$
\begin{aligned}
F8(\vec{x}) &= \sum_{i=1}^{n}(A_i - B_i)^2 \\
A_i &= \sum_{j=1}^{n}(a_{ij}\sin\alpha_j + b_{ij}\cos\alpha_j) \\
B_i &= \sum_{j=1}^{n}(a_{ij}\sin x_j + b_{ij}\cos x_j)\ ,
\end{aligned}
$$

where $n = 30$, $n_\sigma = 30$, and $-\pi \leq x_i^0 \leq \pi$. The $a_{ij}, b_{ij} \in \{-100, \ldots, 100\}$ are random integers, and $\alpha_j \in [-\pi, \pi]$ is the randomly chosen global optimum position. For the matrices $\mathbf{A}, \mathbf{B}$ and the vector $\vec{\alpha}$ we used the values given in Bäck (1996) (pp. 265 267).

No runs found the optimum on this function, resulting in constant speed and success rate curves. Accuracy curves reveal differences between performance for different N's, showing advantageous effects of higher N's, up to approximately 10. The three operators, however, hardly differ in performance and, as the progress curves for $N = 18$ in Fig. 10 indicate, their search behavior is very similar, too.

5. Conclusions

Concerning our first research objective, i.e. finding connections between the characteristics of the objective functions and the usefulness of applying more parents, we can observe the following. With the single exception of Rosenbrock's saddle (F2) it is useful to apply diagonal crossover with "arity" $N > 2$. Looking for particular features of F2 that may cause this deviance let us note that it has the lowest dimensionality ($n = 2$) and the shortest chromosomes (of length 22) as opposed to other functions (200-900 for F3-F8). This makes the disruptiveness of the crossover operators relatively high even for low N's. The other unimodal function in the test set, F1, is apparently so easy to optimize that the GA does not suffer from this effect, but on F2 where the optimum is 'hidden at the bottom of a long bent valley', see Bäck and Michalewicz (1997), this seems to be disasterous.

Our second research objective concerned the identification of the source(s) of increased performance of diagonal crossover when used with more parents. As for the hypothesis H_1, i.e. that increasing the number of crossover points increases performance, observe that N-point crossover did become better for higher N's on all functions of our test suite. Therefore, we accept H_1 and conclude that higher performance *partially* comes from a higher number of crossover points. Explanations for this fact are the better mixing of information, see de

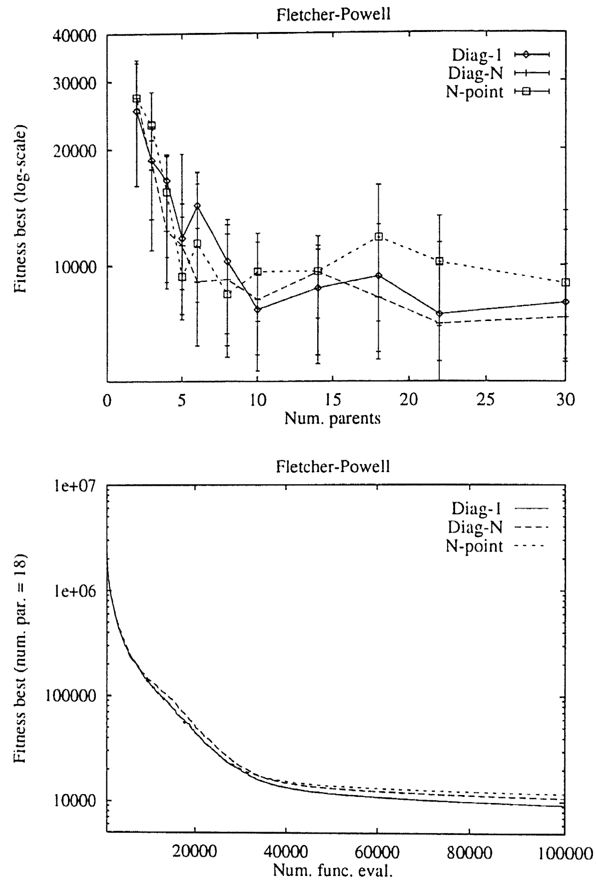

Figure 10. Accuracy (upper part) and progress curves (lower part) for the Fletcher-Powell function

Jong and Spears (1992), and perhaps also the increased macro-mutation-like effects of crossover in case of using higher N's.

Hypothesis H_2 about the advantages of a bigger generational gap is clearly rejected, since the one child and the N children variants of diagonal crossover exhibited the same behavior on each function. Hence, we can conclude that the advantage of applying diagonal crossover with higher N's is not the result of a bigger generational gap in the Steady State GA we use (see Section 2 for discussion).

Recall that our working hypotheses are not mutually exclusive. Accepting H_1 does not imply rejection of H_3, i.e. that better performance for higher N's would *only* come from having more crossover points. In fact diagonal crossover was better than N-point crossover on all but two functions: on Rosenbrock's saddle (F2) and on the Fletcher-Powell function (F8). On the Fletcher-Powell function diagonal crossover was not significantly better than N-point crossover. Such little differences in performance do not clearly justify the acceptence of the hypothesis H_3 on the function F8. There is no clear advantage of using more parents for recombination here. Increased performance for higher N's seems to be the result of the crossovers effect as macro mutation, this effect being intensified by more crossover points. Recall, that F8 spans a very rugged landscape with randomly distributed local optima, which makes it more or less similar to NK-landscapes with relatively high K values. These observations are thus in agreement with earlier conclusions for NK-landscapes, Eiben and Schippers (1996), Hordijk and Manderick (1995), Kauffman (1993), stating that on such surfaces crossover is not useful at all. On Rosenbrock's saddle (F2) diagonal crossover was clearly worse than N-point crossover, besides, the performance of diagonal crossover decreased for increasing N. This behavior is unique on the test suite we use here and at the moment we do not have a solid clarification for it.

According to the above considerations, hypothesis H_3 has to be refined. On quasi-random landscapes, such as F8, increased performance of diagonal crossover for higher N's may occur, but it seems not to be the result of using more parents, i.e. H_3 does not hold. On other types of landscapes (the unimodal F1 and the multimodal, but somewhat regularly shaped F3-F7), diagonal crossover exhibits increased performance when increasing N, and it does outperform N-point crossover, thus confirming H_3. F2 remains an exception, showing that even for unimodal landscapes it is not guaranteed that diagonal crossover will become better when increasing N. Yet, with this exception in mind, we can draw the conclusion that *if* diagonal crossover becomes better for higher N's *then* this improvement is not only the consequence of using more crossover points, but also that of using more parents.

Let us close our conclusions with noting that the greatest gain occurred in the speed of the GA, diagonal crossover is usually faster than 2 parent N-point crossover (if and when). Clearly, if we had set the maximum number of evaluations lower then this difference in speed would also have resulted in differences in success rate and accuracy. Thus, although we definitely do not claim that diagonal crossover is a universally superior operator, we have sufficient evidence to say that it is a sound design heuristics to implement it in a GA and set the number of parents above two.

References

BÄCK, TH. (1996) *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, New York.

BÄCK, TH. and MICHALEWICZ, Z. (1997) Test landscapes. In: Bäck, Th., Fogel, D.B. and Michalewicz, Z., eds., *Handbook of Evolutionary Computation*, B2.7:14–B2.7:20. IOP Publishing Ltd and Oxford University Press.

BERSINI, H., DORIGO, M., LANGERMAN, S., SERONT, G. and GAMBARDELLA, L. (1996) Results of the first international contest on evolutionary optimisation. In: *Proceedings of the 3rd IEEE Conference on Evolutionary Computation*, 611–615. IEEE Press.

BERSINI, H. and SERONT, G. (1992) In search of a good evolution-optimization crossover. In: Männer, R. and Manderick, B., eds., *Proceedings of the 2nd Conference on Parallel Problem Solving from Nature*, 479–488. North-Holland.

BEYER, H.-G. (1966) Toward a theory of evolution strategies: On the benefits of sex- the $(\mu/\mu, \lambda)$ theory. *Journal of Evolutionary Computation*, **3**, 1, 81–111.

BREMERMANN, H.J., ROGSON, M. and SALAFF, S. (1966) Global properties of evolution processes. In: Pattee, H.H., Edlsack, E.A., Fein, L. and Callahan, A.B., eds., *Natural Automata and Useful Simulations*, 3–41. Spartan Books, Washington DC.

DE JONG, K.A. and SPEARS, W.M. (1992) A formal analysis of the role of multi-point crossover in genetic algorithms. *Annals of Mathematics and Artificial Intelligence*, **5**, 1–26.

EIBEN, A.E., VAN KEMENADE, C.H.M. and KOK, J.N. (1995) Orgy in the computer: Multi-parent reproduction in genetic algorithms. In: Morán, F., Moreno, A., Merelo, J.J. and Chacón, P., eds., *Advances in Artificial Life. Third International Conference on Artificial Life*, **929** of *Lecture Notes in Artificial Intelligence*, Springer, Berlin, 934–945.

EIBEN, A.E., RAUÉ, P-E. and RUTTKAY, ZS. (1994) Genetic algorithms with multi-parent recombination. In: Davidor, Y., Schwefel, H.-P. and Männer, R., eds., *Proceedings of the 3rd Conference on Parallel Problem Solving from Nature*, **866** in Lecture Notes in Computer Science, 78–87. Springer-Verlag.

EIBEN, A.E. and SCHIPPERS, C.A. (1996) Multi-parent's niche: n-ary crossovers on NK-landscapes. In: Voigt, H.-M., Ebeling, W., Rechenberg, I. and Schwefel, H.-P., eds., *Proceedings of the 4th Conference on Parallel*

Problem Solving from Nature, **1141** in Lecture Notes in Computer Science, 319–328. Springer, Berlin.

HORDIJK, W. and MANDERICK, B. (1995) The usefulness of recombination. In: Morán, F., Moreno, A., Merelo, J.J. and Chacón, P., eds., *Advances in Artificial Life. Third International Conference on Artificial Life*, **929** of *Lecture Notes in Artificial Intelligence*. Springer, Berlin, 908–919.

KAUFFMAN, S.A. (1993) *Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press.

MÜHLENBEIN, H. (1989) Parallel genetic algorithms, population genetics and combinatorial optimization. In: Schaffer, J.D., ed., *Proceedings of the 3rd International Conference on Genetic Algorithms*, 416–421. Morgan Kaufmann.

MÜHLENBEIN, H. and VOIGT, H.-M. (1995) Gene pool recombination in genetic algorithms. In: *Proceedings of the Metaheuristics Conference*. Kluwer Academic Publishers.

SCHWEFEL, H.-P. (1995) *Evolution and Optimum Seeking*. Wiley, New York.

SCHWEFEL, H.-P. and RUDOLPH, G. (1995) Contemporary evolution strategies. In: Morán, F., Moreno, A., Merelo, J.J. and Chacón, P., eds., *Advances in Artificial Life. Third International Conference on Artificial Life*, **929** of *Lecture Notes in Artificial Intelligence*. Springer, Berlin, 893–907.

VAN KEMENADE, C.H.M., KOK, J.N. and EIBEN, A.E. (1995) Raising GA performance by simultaneous tuning of selective pressure and recombination disruptiveness. In: *Proceedings of the 2nd IEEE Conference on Evolutionary Computation*, IEEE Press, 346–351.

VOIGT, H.-M. and MÜHLENBEIN, H. (1995) Gene pool recombination and utilization of covariances for the Breeder Genetic Algorithm. In: *Proceedings of the 2nd IEEE Conference on Evolutionary Computation*, IEEE Press, 172–177.