When Sharing Fails

Rasmus K. Ursem
EvALife project group, Department of Computer Science
Ny Munkegade, Blg. 540, University of Aarhus
DK-8000 Aarhus C, Denmark
www.evalife.dk ursem@daimi.au.dk

Abstract- Sharing, introduced by Goldberg and Richardson in 1987, is probably one of the most investigated ideas for multimodal optimization. Empirical tests have indicated that sharing is capable of maintaining multiple peaks located simultaneously – a feature that allows a final human selection among the found solutions.

In this paper I present a theoretical argument regarding the performance of sharing. The argument is supported with a series of tests on variants of a simple problem, which is one of Goldberg and Richardson's original test function where a constant is added. The results from these tests indicated that sharing is very sensitive to the range of fitness values.

Finally, three extensions of sharing are proposed and discussed.

1 Introduction

A well-founded heuristic in the field of multimodal optimization is that maintaining high diversity improves the performance of algorithms that aim at reporting multiple peaks simultaneously. High diversity is not only desired in the case of a single objective static objective function, but as well for optimization of dynamic and multiobjective problems. Maintaining diversity in a dynamic environment ensures stable quality of the produced solutions (Ursem, 2000), whereas diversity in multiobjective optimization problems is important for the discovery of the so-called Pareto front[1].

In 1987 Goldberg and Richardson suggested the sharing GA (Goldberg and Richardson, 1987), in which individuals "share" the fitness when they are closer than the predefined distance σ_{share}.

The idea of sharing has, since then, been extended and used in numerous occasions. The original sharing scheme was extended with a mating range in (Deb and Goldberg, 1989). In (Goldberg and Wang, 1997) a coevolutionary variant was suggested and tested. (Miller and Shaw, 1996) describes a sharing GA with dynamic detection of sphere-shaped niches. Sharing is probably now one of the most studied ideas for multimodal optimization.

The central concept of any sharing GA is to downscale the fitness value of individuals with similar genes, i.e., individuals that are close in the search space. In sharing GAs the raw fitness is divided by the so-called sharing factor. The "shared" fitness is often calculated according to the following formula.

$$Fit'(I_i) = \frac{Fit(I_i)}{\sum_{j=1}^{\mu} sh(d(I_i, I_j))} \tag{1}$$

where $Fit'(I_i)$ is the scaled fitness, $Fit(I_i)$ is the raw fitness, μ is the population size, $sh(d)$ is the sharing function, and $d(I_i, I_j)$ is a measure of the distance between individual I_i and I_j. Goldberg and Richardson suggested the following sharing function, which later became widely used.

$$sh(d) = \begin{cases} 1 - (d/\sigma_{share})^{\alpha} & d < \sigma_{share} \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

The parameters σ_{share} and α controls the shape of the sharing function. A common value for α is 1, whereas σ_{share} is more problem dependent.

2 The problem

Goldberg and Richardson tested two functions in their paper from 1987. The functions are one-dimensional maximization problems and are defined according to equation 3 and 4 ($x \in [0, 1]$).

$$F_1(x) = \sin^6(5.1\pi x + 0.5) \tag{3}$$

$$F_2(x) = \sin^6(5.1\pi x + 0.5) \cdot e^{\left(-4\ln(2)\frac{(x-0.1)^2}{0.8^2}\right)} \tag{4}$$

The $F_1(x)$ function is illustrated on figure 1.

In connection with sharing one property of $F_1(x)$ (and also $F_2(x)$) is worth noticing:

The fitness of the minima is zero.

This property is noteworthy because no matter how many individuals are near the maxima they will always

[1]The Pareto-front is the set of optimal compromizes between the objectives.

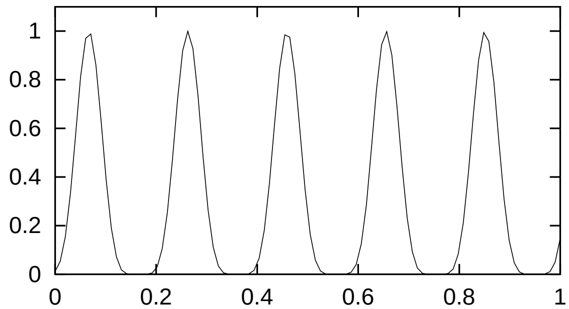

Figure 1: $F_1(x) = \sin^6(5.1\pi x + 0.5)$

have a fitness value greater than zero, which is due to the design of the sharing scheme. These individuals will always have an advantage in selection[2] when they are compared with individuals at the minima. However, if a constant is added to the $F_1(x)$ function, i.e.,

$$F_3(x) = F_1(x) + C \tag{5}$$

then the properties of selection changes substantially.

Sharing was introduced with proportional selection, which was the most widely used selection operator around 1987. A major disadvantage in proportional selection is that the selection pressure is dependent on the range of fitness values. If the fitness ranges between 1 and 2 then the selection pressure is much higher than if the fitness ranges between 1000 and 1001. To illustrate this look at a population with only two individuals and assume that one is a maximal fit individual, while the other is a minimal fit individual. The minimal fit individual has 1/3 chance of survival in the case with fitness values ranging between 1 and 2, whereas it is almost fifty-fifty in the case with fitness between 1000 and 1001.

This problem is expressed in an even higher degree in sharing, because the raw fitness is divided by the sharing factor. Sharing with proportional selection is, in other words, even more sensitive to the fitness range than a standard GA is. In fact, the downscaling of fitness alters the proportional selection operator such that the chance of survival at a maximum is quite related to the ratio between "high fitness area" and "low fitness area" in the search space. The following example will illustrate this. Assume, for the sake of simplicity, that we use the square-shaped sharing function.

$$sh(d) = \begin{cases} 1 & d < \sigma_{share} \\ 0 & \text{Otherwise} \end{cases} \tag{6}$$

Furthermore, assume that $C = 10$ in equation (5), i.e., $F_3'(x) = F_1(x) + 10$. The shape of $F_3'(x)$ is then exactly the same as displayed on figure 1 except that 10 is added to the function. Now, imagine the situation where two individuals are close and near one of the maxima. In this case the scaled fitness of these two individuals can be calculated roughly as

$$Fit'(I) \approx \frac{11}{1+1} = 5.5 \tag{7}$$

where the denominator is 2 because the two individuals are close. These two individuals will *together* appprox-imately get the same slice of the "roulette wheel"[3] as one, three, or twenty individuals that are closer than σ_{share} units. Imagine the simple case with three individuals where two individuals are close to a maximum (fitness≈11) and the third is at a minimum (with no other in range). The chance of picking the low-fit individual instead of one of the high-fit individuals is then roughly 10/21=0.476, or almost 50%. In other words:

Close to random selection!

Sharing is of course less sensitive to this problem if a triangluar sharing function is used (equation (2)). The later presented experiments were conducted with the triangular function, which was to test if the above argument holds in this case as well.

The tournament selection has more or less replaced proportional selection as the most used selection operator. This is because tournament selection is not sensitive to the fitness range or ratio between the fitness of the individuals in the population. However, the downscaling of fitness values poses some problems if tournament selection is used with sharing.

Assume that tournament selection is used with sharing and that it is applied to $F_3'(x) = F_1(x) + 10$. Now, imagine the situation where two individuals are close and near one of the maxima. In this case the scaled fitness of these two individuals can be calculated as in equation (7), i.e., the fitness of an individual near the maximum is circa 5.5. Assume again that a third individual is located near one of the minima such that no other individual is closer than σ_{share} to it. This individual will then have a fitness of approximately 10. During selection the individual at the minimum will win in tournaments when compared to one of the two individuals at the maximum.

> *The advantage of being the only "player" is so great that the isolated individual can win over any pair of individuals that are just fairly close.*

But how close? In the setup above, assuming[4] a tri-

[2] Both proportional and tournament selection.

[3] Proportional selection is also known as roulette wheel selection since the selection procedure corresponds to spinning a roulette wheel.

[4] Values are selected as in Goldberg and Richardsons original

angular function with $\sigma_{share} = 0.1$, $\alpha = 1$, and $C = 10$, then the $sh(d)$ contribution from the other individual in the vicinity only needs to be slightly more than 0.1, i.e., if the distance between the two individuals is less than $0.9 \cdot \sigma_{share}$.

Notice that this problem is not dependent on the constant added to the function; it only depends on the relative fitness between good and bad solutions. A similar argument can be carried out if good solutions have a fitness of 1.0 while bad solutions have a fitness of e.g. 0.75.

Early work on the problem with tournament selection were described in (Oei et al., 1991). Here it was shown that the direct application of tournament selection to sharing had a significant lower performance than a variant Oei, Goldberg, and Chang named *tournament selection with continuously updated sharing*. The shared fitness of the individuals in *tournament selection with continuously updated sharing* is calculated on the basis of what has been copied to the new population at the time of the comparison between the two competing individuals. I.e., the shared fitness of the competing individuals is recalculated in every comparison. This allows more niches to be maintained simultaneously since the first individual in a niche would get an advantage of being the first (sharing factor 1). The selection operator favors individuals discovering new niches over individuals located in already discovered niches. This method was used in the later described experiments on tournament selection.

3 Experiments

The experimental setup is almost the same as in the original work, i.e.,

$$\begin{aligned} popsize &= 50 \\ p_c &= 0.8 \\ \#bits &= 30 \\ \#iterations &= 100 \end{aligned}$$

Tests were performed on

$$F_3(x) = F_1(x) + C$$

with C=0,1,2,5, and 10. The only differences between the experiments presented here and in (Goldberg and Richardson, 1987) are (i) the population is randomly initialized in every run, and (ii) 100 runs were performed for each C-value.

Four set of experiments were conducted. Two with proportional selection where $p_m = 0.0$ and $p_m = 0.03333$. The first corresponds to Goldberg and Richardsons orginal experiment, while the second was performed since a low mutation rate rendered more stable results for the case $C = 0$. The last two experiments were conducted using *tournament selection with continuously updated sharing* and again $p_m = 0.0$ and $p_m = 0.03333$.

The bar graphs (figure 2 and 3) were produced by dividing the search space into 50 intervals of length 0.02 and count the number of individuals in each interval during the last 5 iterations of each run. The counts were then averaged for the 100 runs and scaled to a percentage. Two percent corresponds to one individual ($popsize = 50$). The first row (in both figures) contains graphs for the case $C = 0$, second row $C = 1$, third row $C = 2$, fourth row $C = 5$, and fifth row $C = 10$.

The results from the four experiments are quite alike and will thus be handled all as one. The first row ($C = 0$) corresponds to Goldberg and Richardsons original experiment. The experiments here support their experiment in all four cases, i.e., the individuals are located near the peaks.

The second row contains the graphs for the case $C = 1$. Here several individuals are located near the minima. However, the peaks are still clearly visible on the graphs.

In case $C = 2$ and $C = 5$ are the peaks still detectable but in the case $C = 10$ are the individuals so scattered that approximately one individual is in each interval – a situation similar to random initialization.

Preliminary experiments were conducted using tournament selection *without* continuously updated sharing, i.e. standard tournament selection where the scaled fitnesses are calculated before the selection. These experiments supported the results from (Oei et al., 1991), i.e., that the naive application of tournament selection to sharing had a worse performance than the *tournament selection with continuously updated sharing*.

4 Extensions to sharing

From the previous sections it seem that sharing is heavily dependent on the fact that the minima have a fitness of zero. An obvious idea that comes to mind is to rescale the fitness function such that this property is expressed. For instance, the best and worst raw fitness in the current population provides the necessary means to rescale the fitness of all individuals to the interval between zero and one. Adding a constant to the rescaled fitness allows easy control of the selection pressure, which will enable the algorithm to sustain individuals at the lowest peak. These individuals would otherwise get a fitness near zero.

Another approach would be to use a fixed fitness transformation to alter the fitness function such that the difference between good solutions and less good solutions is enhanced. A linear transformation will probably be insufficient, because it would require extensive knowledge of the range of possible fitness values. Furthermore, a linear transformation does not enhance the

experiment.

relative difference between good and less good solutions. However, an exponential fitness transformation does not have these drawbacks. The fitness of an individual could be calculated according to

$$Fit'(I_i) = \frac{\exp(\sigma_{trans} \cdot Fit(I_i))}{\sum_{j=1}^{\mu} sh(d(I_i, I_j))} \quad (8)$$

where σ_{trans} controls the shape of the exponential function. In addition, the exponential fitness transformation solves the problem with negative values of the objective function[5]. The drawback is that σ_{trans} will probably be quite dependent on the shape and height of the peaks.

A third idea is to subtract the sharing factor instead of dividing with it. The subtracted value p could, for instance, be proportional to the niche count m, i.e., $p = m\sigma_{capacity}$. The advantage is that the algorithm is independent of the range of fitness values. The drawback is that $\sigma_{capacity}$ is heavily dependent on the difference between the fitness of the local peak and its immediate neighborhood. For $F_1(x)$ will a capacity of $\sigma_{capacity} = 0.1$ allow a niche capacity of roughly 10 individuals, because the difference between maxima and the minima is approximately 1.

5 Alternatives to sharing

Over the years several algorithms for locating multiple optima have been proposed. One of the first was De Jong's crowding algorithm (De Jong, 1975), in which the offspring replaces the most similar individual of a randomly drawn small subset. Mahfoud suggested a variant called deterministic crowding (Mahfoud, 1992) where the offspring competes with the parents. Petrowski introduced the clearing operator (Petrowski, 1997). Clearing sorts the individuals according to fitness and determines a number of local winners. All the losers are assigned a fitness of zero.

Another approach is used by the so-called search space division[6] algorithms. Here the population is divided into a (variable) number of subpopulations each corresponding to different potential peaks in the search space. The forking GA was suggested by Tsutsui and Fujimoto ((Tsutsui and Fujimoto, 1993), see (Tsutsui et al., 1997) for a complete description). The forking GA divides the search space by detecting genetic similarities (genotypic forking) or by measuring similarity in the search space (phenotypic forking). Oppacher and Wineberg introduced the shifting balance GA (Oppacher and Wineberg, 1999), which enforces the search space division by a special selection operator that drives the subpopulations apart when they overlap. Finally, the multinational GA was proposed in (Ursem, 1999). Here the subpopulations are separated by a hill-valley detection scheme that groups the individuals according to the topology of the fitness landscape, i.e., detection of valleys in the fitness landscape between points in the search space.

[5]If the objective function is used as fitness function and it is negative the individuals actually benefit from clustering close together.

[6]The term "search space division" constitutes the idea of actively arranging the individuals into non-overlapping subpopulations and restrict mating to these subpopulations.

6 Conclusions

In this paper I presented a theoretical argument regarding the performance of sharing in the form used until now. The speculations were verified on variants of one of the test problem introduced in the original work on sharing. The experiments clearly showed that sharing is extremely sensitive to the range of fitness values defined by the objective problem. The performance of sharing seems to depend on whether or not the fitness value of the less wanted solutions[7] is close to zero – a constraint hard to fulfill if the fitness values of the unwanted solutions are too different.

Three simple extensions to sharing were suggested. Extensive testing on a large test suite are needed to reveal whether or not these ideas can improve the overall performance of sharing.

Acknowledgements

The author would like to thank Thiemo Krink, Brian H. Mayoh, René Thomsen, Peter Rickers, Lars Kroll Kristensen, Mikkel T. Jensen, and Zbigniew Michalewicz for valuable comments and suggestions.

References

[De Jong, 1975] De Jong, K. A. (1975). *An Analysis of the Behavior of a Class of Genetic and Adaptative Systems.* PhD thesis, University of Michigan, Ann Arbor.

[Deb and Goldberg, 1989] Deb, K. and Goldberg, D. E. (1989). An Investigation of Niche and Species Formation in Genetic Function Optimization. In Schaffer, editor, *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 42–50.

[Goldberg and Richardson, 1987] Goldberg, D. E. and Richardson, J. (1987). Genetic Algorithms with Sharing for Multimodal Function Optimization. In Grefenstette, editor, *Genetic Algorithms and their Applications (ICGA'87)*, pages 41–49.

[Goldberg and Wang, 1997] Goldberg, D. E. and Wang, L. (1997). Adaptive Niching Via Coevolutionary Shar-

[7]Local minima/maxmima.

ing. Technical Report 97007, Illinois Genetic Algorithms Laboratory, University of Illinois at Urbana-Champaign.

[Mahfoud, 1992] Mahfoud, S. (1992). Crowding and preselection revisited. Technical Report 92004, Illinois Genetic Algorithms Laboratory (IlliGAL).

[Miller and Shaw, 1996] Miller, B. L. and Shaw, M. J. (1996). Genetic Algorithms with Dynamic Niche Sharing for Multimodal Function Optimization. In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pages 786–791.

[Oei et al., 1991] Oei, C. K., Goldberg, D. E., and Chang, S.-J. (1991). Tournament Selection, Niching, and the Preservation of Diversity. Technical Report 91011, University of Illinois at Urbana-Champaign (IlliGAL).

[Oppacher and Wineberg, 1999] Oppacher, F. and Wineberg, M. (1999). The Shifting Balance Genetic Algorithm: Improving the GA in a Dynamic Environment. In Banzhaf et al., editors, *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 504–510.

[Petrowski, 1997] Petrowski, A. (1997). A New Selection Operator Dedicated to Speciation. In Bäck, editor, *Proceedings of the 7th International Conference on Genetic Algorithms*, pages 144–151.

[Tsutsui and Fujimoto, 1993] Tsutsui, S. and Fujimoto, Y. (1993). Forking Genetic Algorithm with Blocking and Shrinking Modes (FGA). In Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 206–213.

[Tsutsui et al., 1997] Tsutsui, S., Fujimoto, Y., and Ghosh, A. (1997). Forking Genetic Algorithms: GAs with Search Space Division Schemes. *Evolutionary Computation*, 5:61–80.

[Ursem, 1999] Ursem, R. K. (1999). Multinational Evolutionary Algorithms. In Angeline et al., editors, *Proceedings of the Congress of Evolutionary Computation*, volume 3, pages 1633–1640.

[Ursem, 2000] Ursem, R. K. (2000). Multinational GAs: Multimodal Optimization Techniques in Dynamic Environments. In *Proceedings of the Second Genetic and Evolutionary Computation Conference*.

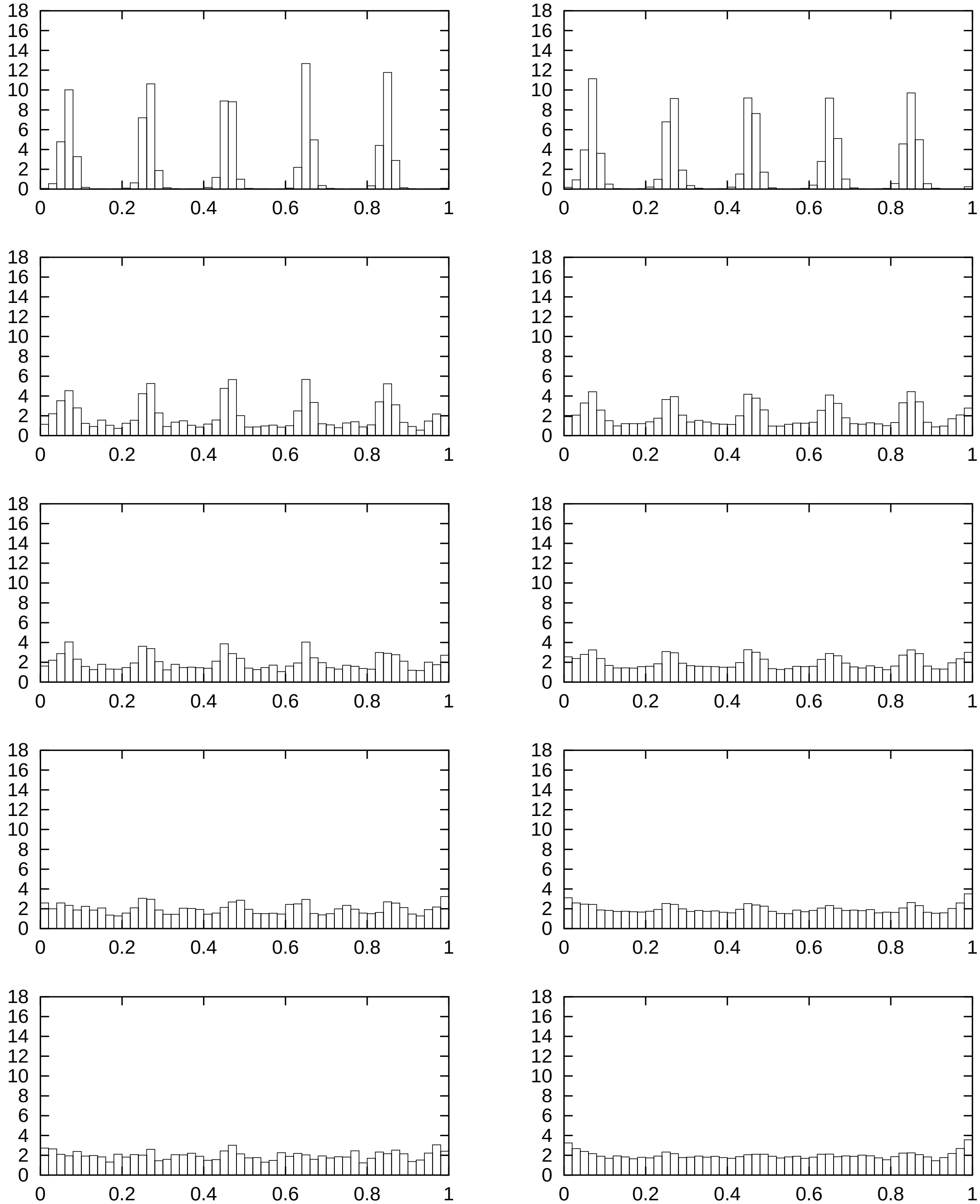

Figure 2: Results for the experiments with proportional selection. Left column: $p_m = 0.0$, right column: $p_m = 0.03333$. First row: $C = 0$ (as the original experiment), second row: $C = 1$, third row: $C = 2$, fourth row: $C = 5$, fifth row: $C = 10$. The y-axis is the percentage of individuals in the corresponding x-range. Two percent corresponds to one individual ($popsize = 50$).

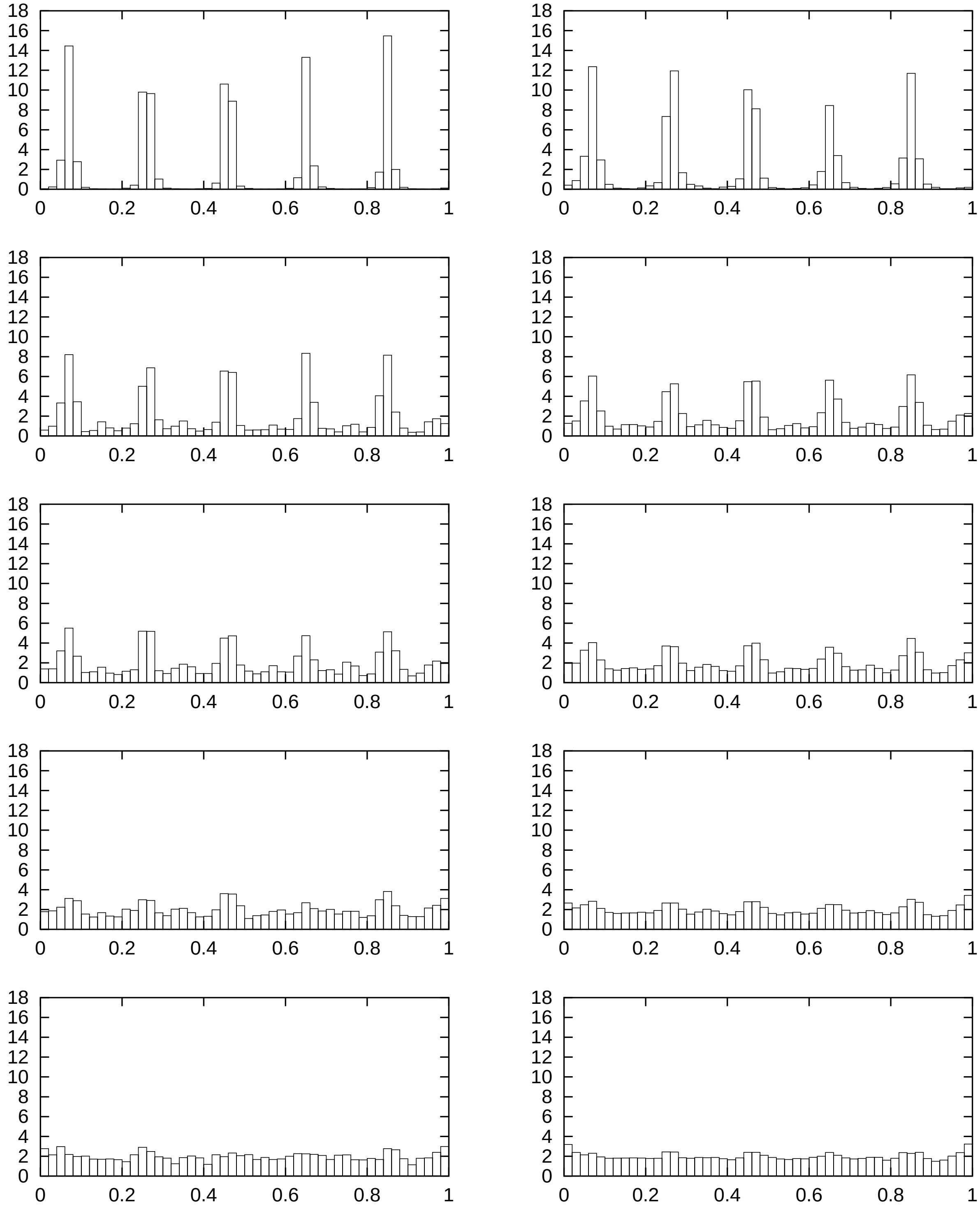

Figure 3: Results for the experiments with tournament selection. Left column: $p_m = 0.0$, right column: $p_m = 0.03333$. First row: $C = 0$ (as the original experiment), second row: $C = 1$, third row: $C = 2$, fourth row: $C = 5$, fifth row: $C = 10$. The y-axis is the percentage of individuals in the corresponding x-range. Two percent corresponds to one individual ($popsize = 50$).