4

Exploration and Exploitation

> A successful application of an optimizer resides in the well-found trade-off between exploration and exploitation. So, we are continuously searching for the best equilibrium between them. In this chapter we pass to the analysis of the differentiation operation and equally to the study of the control parameter influence. In order to make a better choice of strategy I propose calculating an indicator of the strategy diversity,[1] its exploration capacity. Also, I shall show that differentiation is the first step to the general operator integrating mutation and crossover, where mutation provides the needed diversity of the population and crossover assures the capacities to survive. Moreover, in this chapter I expose my studies consecrated to the control parameters and their tuning. This results in practical recommendations for using differential evolution.

When we speak about evolutionary algorithms — GA, ES, DE, or others — we always expect to find the global optimum, but...

> *...the ability of an EA to find a global optimal solution depends on its ability to find a right relation between exploitation of the elements found so far and exploration of the search space....* [Bey98]

Thus, the successful application of the method consists in the choice of the optimal exploration/exploitation union. As is well known, the excessiveness of exploration leads to the global optimum with a high probability, but critically slows down the convergence rate. On the other hand, the excessive exploitation quickly results in local optima.

The capability of genetic operators to control exploration/exploitation balance as well as their relative importance has been discussed for many decades. Some groups of scientists believe in mutation-selection superiority, others concentrate themselves on crossover power. But I make an effort to be impartial to these opinions and elicit the advantages from both points of view.

[1] The diversity measures the proportion of the surveyed space of solutions.

4.1 Differentiation via Mutation

The strategies, which use objective function values to create the trial individual, accomplish an exploitation function. These are *dir* and *dir-best* groups. The diversity in this case decreases twice, so in order to maintain it at a required level it is necessary to increase the population size and/or the number of extracted individuals.

The fact of random choice of parents for a trial individual itself answers for exploration capabilities. Besides the population size and the type of strategy, exploration efficiency can be controlled by the differentiation constant F as well [Zah01, LL02b, Š02].

To the present day, it was considered (disruption and construction theories) that mutation cannot completely fulfill the functions of crossover and vice versa [Spe93]. Mutation perfectly creates a random diversity, but it cannot execute the construction function well. Crossover can show preservation, survival, and construction, but often it cannot achieve a desirable diversity. Thus, the EC community was looking forward to the one general operator that could integrate mutation and crossover functions as well as any variations between them. Differentiation in the DE algorithm is the first step on the road to such an operator. It does not fall under the influence of accepted disruption theory providing needed diversity and, at the same time, it luxuriously preserves survival capabilities. Let discuss it in more detail.

Differentiation is the first step to the general operator.

There is no disruption effect for differentiation! Disruption rate theory estimates the probability that an evolutionary operator will disrupt a hyperplane sample, in other words, the probability that individuals within a hyperplane will leave that hyperplane [Spe93]. Let all individuals of a population X_i belong to hyperplane H. Hence, β and δ are always on H. Therefore, $\omega = \beta + F \cdot \delta$ will belong H too. That is, there is no combination of individuals on the hyperplane that makes the trial individual leave this hyperplane. This means a good survival capability of differentiation usually inherent to crossover.

4.2 Crossover

The principal role of crossover is as a construction. There is no such mutation that can achieve higher levels of construction than crossover [Spe98]. Just as selection exploits objective function values, crossover exploits genetic information. Moreover, crossover furnishes the high diversity of a population.

Convinced of the power of crossover I would like to make a point about applying it to DE.

Videlicet, in the cases when we use the strategies with a direction analysis (*dir* and *dir-best* groups) crossover operation becomes unnecessary, because it spoils a trial individual inducing the noise. In other words, when we choose the directed strategies, it is supposed that we want to imitate the gradient function, that is, to make the steps close to the gradient direction. If we use crossover, the gene's exchange between the trial and target individuals would perturb the desired direction in most cases.

Furthermore, note that differentiation by itself is capable of executing the both functions (exploration/exploitation) simultaneously. So, if we guaranteed sufficient exploration (diversity of population), then the crossover operation would be superfluous. Thus we could eliminate it and thereby reduce computing time as well.

4.3 Analysis of Differentiation

The structure of the DE algorithm is similar to that of genetic algorithms: concepts of mutation and crossover are repeated here. In addition, DE integrates the ideas of self-adaptive mutation specific to evolution strategies. Namely, the manner of realization of such a self-adaptation has made DE one of the most popular methods in evolutionary computation. We examine it in detail.

Originally, two operations were distinguished: differential mutation and continuous recombination [Pri99]. Differential mutation was based on the strategy $\omega = \xi_3 + F \cdot (\xi_2 - \xi_1)$ and required at least three randomly extracted individuals. A continuous recombination was in need of only two individuals, $\omega = \xi_1 + K \cdot (\xi_2 - \xi_1)$. Price emphasized the different dynamic effects of these operations. In the case of a continuous recombination the trial individual ω places only on the line created by its parents ξ_1, ξ_2. This compresses a population. In the case of a differential mutation the difference vector $(\xi_2 - \xi_1)$ is applied to an independent individual. And it is similar to the Gaussian or Cauchy distribution used in ES, that makes no reference to the vector to which it is applied. It does not compress a population. Founded on such an inference several strategies were proposed [Sto96a].

Recently, in 2004, a new vision of these operations was discovered (see [FJ04d] or Chapter 3). A new principle of strategy design (see [FJ04g] or Section 3.2) was introduced, which synthesizes the previous two operations by one unique formula and accentuates population diversity. Now, all strategies are described by two vector terms: difference vector δ and base vector β (2.8). The difference vector provides a mutation rate term (i.e., a self-adaptive noise), which is added to a randomly selected base vector in order to produce a trial individual. The self-adaptation results from the individuals' positions. During the generations the individuals of a population occupy more and more profitable positions and regroup themselves. So, the difference vector decreases (automatically updates) each time the individuals fit local or global optima.

The strategies have been classified into four groups by information that they use to "differentiate" the actual individual (rand/dir/best/dir-best). Each group represents a proper method of search (random/directed/local/hybrid). Hence, the key to ensure a required diversity of a population is not only the dynamic effect of the operations, but, to a greater extent, the number of randomly extracted individuals k needed to create a strategy.

We look at differentiation now from a combinatorial point of view. Usually, population size NP and the constant of differentiation F are fixed. Thus NP individuals are capable of producing potentially $\Theta(k)$ different solutions. We refer to $\Theta(k)$ as a diversity characteristic of a strategy, whereas the method of using of these individuals reflects strategy dynamics (see Sections 3.2 and 3.4 or [FJ04b]). We shall give an estimation of diversity. Let the individuals be extracted from the population one after another, so the upper diversity bound can be evaluated in the following way (see Fig. 4.1),

$$\overline{\Theta}(k) = \prod_{i=1}^{k}(NP - i)\,. \tag{4.1}$$

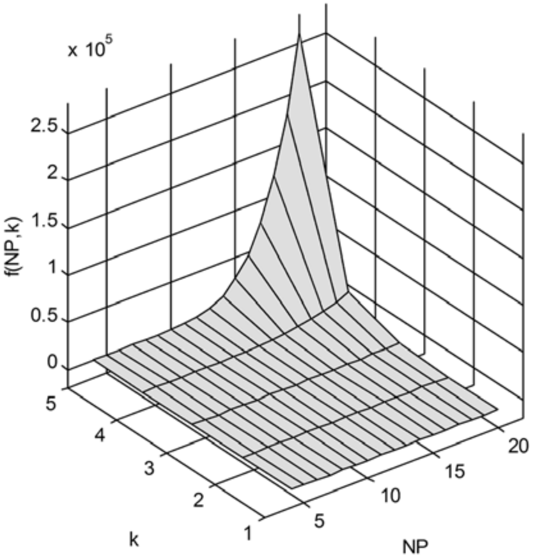

Fig. 4.1. The upper diversity bound.

The strategy (difference vector) is created by calculating the barycenters of two sets of individuals. The general differentiation formula can be rewritten as

$$\omega = \beta + F \cdot (Bar(set_2) - Bar(set_1))\,. \tag{4.2}$$

These sets, besides randomly extracted individuals (n_1 in the first set and n_2 in the second set), may also include the target and the current best individuals. Because of such a generalization, the extraction order of individuals forming a barycenter is not important, thus the diversity of population decreases in $n_1! \cdot n_2!$ times, where $n_1 + n_2 = k$. Moreover, if the directed or hybrid strategies are used, then the diversity still drops down twice. Therefore we introduce a direction factor

$$dir = \begin{cases} 2 & \text{if RAND/DIR or RAND/BEST/DIR} \\ 1 & \text{if RAND or RAND/BEST} \end{cases} \tag{4.3}$$

Consequently, the exact diversity estimation is equal to:

$$\Theta(k) = \frac{\prod_{i=1}^{k}(NP - i)}{dir \cdot n_1! \cdot n_2!} \,. \tag{4.4}$$

It is obvious that a high diversity slows down the convergence rate, whereas a low one results either in stagnation or premature convergence. Thus some balance is needed. By controlling the number of randomly extracted individuals (or more precisely a strategy type) we can easily provide the required diversity of population (see Section 3.4).

Practical remark: As you already know, the diversity function depends on the number of individuals used in a strategy and the size of a population. If the number of individuals in a strategy surpasses 7 then the diversity of a strategy becomes enormous and consequently the exploration would be excessive. So, in practice, it is reasonable to use not more than 3, 4, or 5 individuals to give sufficient exploration. This is a practical compromise between computing time and quality of the search space exploration.

4.4 Control Parameters

The goal of control parameters is to keep up the optimal exploration/exploitation balance so that the algorithm will be able to find the global optimum in the minimal time. Practical tests show that within the search process the diversity of a population (its exploration capabilities) usually goes down more rapidly than we would like. Thus, one of the ways to provide good control is to retain the desired diversity level.

4.4.1 Diversity Estimation

During the last years different manners of the diversity estimation were proposed in the literature. I shall represent some of them here.

Expected Population Variance

One of the exploration power measures is the population variance [BD99]

$$\mathrm{Var}(\mathbb{P}) = \overline{X^2} - \overline{X}^2 \,, \tag{4.5}$$

where $\overline{X}$ is a population mean and $\overline{X^2}$ is a quadratic population mean. So, if $\mathrm{Var}(\mathbb{P}^0)$ is an initial population variance, then after several generations the expected population variance can be estimated as a function of the control parameters $\Omega = \Omega(F, Cr, NP, k)$ [Zah01].

$$E(\mathrm{Var}(\mathbb{P}^g)) = \Omega^g \cdot \mathrm{Var}(\mathbb{P}^0) \,. \tag{4.6}$$

The comparisons of the real and theoretical results confirm the likelihood of such an estimation. To retain the given diversity it is necessary for the transfer function Ω to be a little more than or at least equal to 1: $\Omega \geq 1$.

Average Population Diversity

In the work [Š02] a direct measure of average population diversity was introduced.

$$div(g) = \frac{\sum_{i=1}^{NP} \sum_{j=i+1}^{NP} \frac{|X_i(g) - X_j(g)|}{H-L}}{2 \cdot D \cdot (NP-1) \cdot NP} \,. \tag{4.7}$$

It represents for each generation an average normalized distance between the individuals of the population.

Mean Square Diversity

Another direct way to estimate diversity is to use the mean square root evaluation for the population as for its objective function [LL02a].

$$\begin{aligned} P_{div}^g &= \frac{1}{k_p} \sqrt{\frac{1}{NP} \sum_{i=1}^{NP} \sum_{j=1}^{D} (x_{i,j}^g - x_{i,j}^{g-1})^2} \\ F_{div}^g &= \frac{1}{k_f} \sqrt{\frac{1}{NP} \sum_{i=1}^{NP} (f_i^g - f_i^{g-1})^2} \,, \end{aligned} \tag{4.8}$$

where k_p, k_f compress P_{div}^g, F_{div}^g into the interval $[0, 1]$. This method requires an additional memory source both for the population and the vector of objective functions of the previous generation.

P-Measure

There is a simpler and, perhaps, more practical way to estimate population diversity (see Chapter 5). $P(population)$-measure is a radius of population, that is, an Euclidean distance between the center of population O_p and the farthest individual from it.

$$P_m = \max \|X_i - O_p\|_E, \qquad i = 1, \ldots, NP\,. \tag{4.9}$$

4.4.2 Influence of Control Parameters

Constant of Differentiation

The constant of differentiation F is a scaling factor of the difference vector δ. F has considerable influence on exploration: small values of F lead to premature convergence, and high values slow down the search. I have been enlarging the range of F to the new limits $F \in (-1, 0) \cup (0, 1+]$ (see Subsection 3.2.5). Usually, F is fixed during the search process. However, there are some attempts to relax this parameter. Relaxation significantly raises the covering of the search space and also partially delivers us from the exact choice of F. Among the relaxations we can outline $F = N(0, F)$, $N(F, \sigma)|_{\sigma \ll F}$, and $N(F, F)$ with a normally distributed step length and the same variants with uniform distribution.

Constant of Crossover

The constant of crossover reflects the probability with which the trial individual inherits the actual individual's genes. Although using Crossover makes the algorithm rotationally dependent (Appendix D and [Sal96, Pri99]), crossover becomes desired when we know the properties of an objective function. For example, for symmetric and separable functions $Cr \approx 1 - 1/D$ is the best choice; for the unimodal (or quasi-convex) functions a good choice is crossover with the best individual. Moreover, small values of Cr increase the diversity of population. To put it differently, the number of potential solutions will be multiplied by the number of vertices of a D-dimensional hypercube built on the trial and target individuals.

Size of Population

The size of population NP is a very important factor. It should not be too small in order to avoid stagnation and to provide sufficient exploration. The increase of NP induces the increase of a number of function evaluations; that is, it retards convergence. Furthermore, the correlation between NP and F may be observed. It is intuitively clear that a large NP requires a small F; that is, the larger the size of a population is, the more densely the individuals fill the search space, so less amplitude of their movements is needed.

Type of Strategy

The strategy can be characterized by the number of randomly extracted individuals k and the dynamic effect resulting from the manner of their use. k controls the diversity, whereas the way to calculate the trial individual directly reflects the dynamics of exploration. A small k makes the strategy a slack one. A big k slows down the convergence rate because of both the excessive diversity and towering complexity of differentiation.

4.4.3 Tuning of Control Parameters

The effective use of an algorithm requires the tuning of control parameters. And this is a time-consuming task. However, the parameter tuning may be replaced by the parameter control [EHM99].

Three types of parameter control are distinguished.

1. *Deterministic control:* parameters are followed by a predefined deterministic law; there is no feedback information from the search process.
2. *Adaptive control:* parameters depend on feedback information.
3. *Self-adaptive control:* parameters depend on the algorithm itself; they are encoded into it.

Deterministic Control

For the first time the deterministic control of the population size has been introduced using the energetic selection principle (Chapter 8). The population is initialized by a huge number of individuals; then an energetic barrier (deterministic function that depends on the generation number) is applied to reduce the population to a normal size. This method leads to global convergence and increases its rate.

Next, the law switching from one type of strategy to another can be implemented. In such a way both the number and type of used individuals are controlled. Switching results from a predefined switch-criterion that depends, for instance, on the relative difference $(f_{\max} - f_{\min})$.

Adaptive Control

Two methods of adaptive control are distinguished.

1. Refresh of population
2. Parameter adaptation

The refresh of population [Š02] is realized either by replacement of “bad” individuals or by injecting individuals into the population. Both methods increase the diversity. The refresh can be aimed at exploration of new regions of the search space as well as for convergence rate improvement. It repeats periodically or each time when the population diversity reaches a critical level.

The parameter adaptation entirely obeys the state of population. The feedback information calculated on the basis of this state modifies the control parameter according to a control law.

By now, two variants of adaptation have been proposed.

- The first one is a fuzzy control that adjusts the constant of differentiation F. The input signal for the fuzzy system is computed from (4.8). Membership functions and fuzzy rules are established based on expert knowledge and previous tests. Notice that diversity evaluation, fuzzification/defuzzification, and execution of fuzzy rules are time-consuming operations and their complexity might be comparable with one DE generation. Thus, it is always necessary to estimate the relative efficiency of this method.
- The second one is an adaptation based on the theoretical formula of the expected population variance (4.6) [Zah02]. Also, the parameter F is adjusted. But this parameter becomes unique for each gene of the individual; that is, each gene has its own parameter value. The adaptation happens each generation. It is less complex than the previous one ($O(NP \cdot D)$) and does not modify the complexity order of one generation. Such an adaptation prevents premature convergence, but does not ensure the best convergence rate. Moreover, it does not depend on the objective function, so the introduction of supplementary information would be desirable.

Self-Adaptive Control

The work [Abb02] first proposed self-adaptive crossover and differential mutation. Also, separate parameters were proposed for each individual as well as for differential mutation as for crossover. The self-adaptation scheme repeats the principle of differential mutation: $r = r_{\xi_3} + N(0,1) \cdot (r_{\xi_2} - r_{\xi_1})$, $r \in \{F, Cr\}$. The given adaptation was used for multiobjective Pareto optimization and the obtained results outperform a range of state-of-the-art approaches.

4.5 On Convergence Increasing

Actually, we emphasize three trends of convergence improvement.

1. Localization of global optimum
2. Use of approximation techniques
3. Hybridization with local methods

All these trends represent a pure exploitation of available information. They elevate the intelligence of the algorithm in order to improve its convergence. The main purpose of the prescribed improvement is an ability to solve large-scale nonlinear optimization problems.

Localization

The energetic selection principle (Chapter 8) is a particular case that illustrates a fast localization of the global optimum. An initialization by a large population helps to reveal promising zones; then a progressive reduction of the population locates the global optimum. At the end, local techniques can be used.

Approximation

The deterministic replacement of “bad” individuals by “good” ones is one of the ideas to ameliorate the convergence. Let the good individuals be created by approximation methods. For example, we construct a convex function regression on the basis of the best individuals of the population. Then, the optimum of this regression will replace the worst individual. Here, there are lots of regression techniques that could be applied. For instance, a more recent and promising one is support vector machines (Chapter 9). The main emphasis is made on choosing an appropriate kernel function, which considerably influences the quality of approximation.

Local Methods

The most traditional idea is to use the population-based heuristics as “multistarts” for deterministic optimizers. The positive results were demonstrated by hybridizing DE with the L-BFGS method [AT02]. This hybridization proves to be more efficacious for large-scale problems than for small ones.

Problems

4.1. What do we mean by exploitation and exploration when speaking about evolutionary algorithms?

4.2. What advantages and disadvantages has an excessive exploration? And an excessive exploitation?

4.3. It is well known that the role of genetic operators is to control the balance between exploration and exploitation. If you had a choice between mutation and crossover, what would you prefer? And why?

4.4. Choose from Chapter 3 four different strategies (one strategy from each group) and explain how the strategies realize the functions of exploitation and/or exploration.

4.5. What do we mean by the diversity of population? Analyse how the diversity of population changes as exploitation increases?

4.6. Analyse the efficiency of exploration when the constant of differentiation F increases.

4.7. What is the general operator? Could you consider the operation of differentiation as the general operator? Explain your point of view.

4.8. Explain the disruption effect. Does the differentiation operator possess this effect?

4.9. Suppose n individuals in n-dimensional space E^n are linearly dependent vectors. Then, among them, there exists r linearly independent vectors forming the basis in the subspace $E^r \subset E^n$. Let the optimum $Opt \in E^n$ be outside of subspace E^r, that is, there are no decompositions on basis vectors. The DE algorithm implements only differentiation and selection (without crossover). Is the found solution X^* the optimum Opt? Write your arguments and give an explaining sketch.

4.10. What properties should crossover have? Enumerate at least three properties and give an example for each of them.

4.11. How does the exploitation property appear in crossover?

4.12. In which cases does crossover become useless and may be even harmful?

4.13. Due to what does the self-adaptation of difference vector δ occur?

4.14. How, in theory, does one estimate the diversity of population from a combinatorial standpoint? Take, from Chapter 3, any three strategies and calculate the diversity according to the formula (4.4) of Chapter 4.

4.15. Test the strategies selected for problem (4.14) using any test function you have. Analyse the influence $\Theta(k)$ on the precision of the found solutions and the computing time spent to obtain them. Make a written deduction.

4.16. Why are the control parameters necessary?

4.17. What empiric methods for diversity estimation do you know? Enumerate at least four methods and implement one of them for choice.

4.18. Plot experimental curves (diversity from generation) for the strategies chosen in problem (4.14) and the function used in problem (4.15). Explain the obtained results.

4.19. What is the relaxation of F? Are there any advantages of relaxation? Give some examples of relaxation. Does relaxation have some drawbacks?

4.20. Given a test function, the so-called Salomon function,

$$\begin{aligned} f(X) &= -\cos(2\pi\|X\|) + 0.1 \cdot \|X\| + 1\,, \\ \|X\| &= \sqrt{\sum_{i=1}^{D} x_i^2}\,, \quad -100 \le x_i \le 100\,, \\ f(X^*) &= 0\,,\ x_i^* = 0\,,\ VTR = 1.0 \times 10^{-6}\,. \end{aligned}$$

Plot this function for $D = 2$. Make two experiments: the first for the fixed F and the second for the relaxed F. Compare the results and show at least one advantage and one disadvantage of the relaxation.

4.21. In which cases is crossover definitely necessary? Give concrete examples.

4.22. For any problem you choose, initialize the population as described in problem (4.9). Find the optimal solution without using the crossover operation. Then, add the crossover operation and watch whether the new-found optimal solution is changed? Make tests with different values of crossover. Which value of crossover is the best for your case?

4.23. What chances do you take when the population size is too small? Demonstrate on an example the stagnation effect of the algorithm. For the demonstrated example, plot a graph of the time (generations), needed to find the optimal solution (VTR) from the size of the population NP.

4.24. Choose arbitrarily one of four groups of strategies. Test several strategies of this group on either your favorite problem or any known test function. Plot curves of convergence for these strategies. Analyse how the convergence of the algorithm depends on the number of randomly extracted individuals needed for one or another strategy? Did you use, for validity of results, one and the same initial population for all your tests?

4.25. Why do we need to adjust the control parameters?

4.26. What is the difference between tuning of control parameters and the parameter control? What three types of parameter control do you know?

4.27. Think out your own method of deterministic control, implement it and test. Estimate its efficiency.

4.28. What kinds of adaptive control do you know?

4.29. Elaborate your own method of adaptive control for one of four parameters (F, Cr, NP, k). Implement it and test. Estimate its efficiency.

4.30. Explain in short what is the self-adaptive control.

4.31. Think out your own version of self-adaptive control. Test it and estimate how efficient your version is.

4.32. For optimization of a test function, the so-called Schwefel's function,

$$f(X) = -\frac{1}{D}\sum_{i=1}^{D} x_i \cdot \sin(\sqrt{|x_i|}) \ , \qquad -500 \leq x_i \leq 500 \, ,$$

$$f(X^*) = -418.983 \ , \qquad x_i^* = 420.968746 \ , \qquad VTR = 0.01 \, ,$$

use the DE algorithm with three control parameters (F, Cr, NP). Your task is to optimize these parameters in order to ameliorate the convergence of the algorithm. For this, use an algorithm of global optimization (either DE or another). Plot the Schwefel's function for $D = 2$. Write the program and find the optimal control parameters.

4.33. What three trends of convergence improvement do you know? To what are these improvements usually attributed? Write a short explanatory paragraph.