Complex Systems **10** (1996) 311–319

Convergence in Simulated Evolution Algorithms

Olivier François
Laboratoire de Modélisation et Calcul,
Institut IMAG, Université de Grenoble,
BP 53, 38041 Grenoble cedex 9, France

Abstract. New simulated evolution algorithms are presented for global minimization. The main feature of these algorithms is to couple the standard mutation and selection operators into a single operator. Exponential probabilities are also introduced so as to stress the relevance of statistical mechanics arguments. The work follows the stream of large deviations techniques developed to analyze the simulated annealing and genetic algorithms.

1. Introduction

The standard approach to optimization is to formulate a cost function that translates the value of a decision and iteratively improve this value by selecting among available alternatives. Optimization has a long history and many classes of problems have been considered in the past. The methods in this paper use randomness as a dominant mechanism. Consequently, the corresponding search procedures are expected to be robust and not sensitive to irregular cost functions. We shall consider new evolutionary algorithms. Each maintains a population of trials solutions, imposes random changes to those solutions, and incorporates the use of selection to determine which solutions to maintain in future generations and which to remove from the pools of trials [2, 7, 10, 11]. Following is an informal presentation of these procedures.

- The problem to be addressed is captured in a positive cost function f which is defined on a finite set E. In [11], it is suggested that all solutions should be represented as binary strings. In such a case, the set E would be equal to $\{0,1\}^l$ for some integer $l \geq 0$.
- A population of trial solutions is initialized and constrained to be in E.
- The new generation is randomly divided into two complementary subsets. Then, selection acts on the current population and the offspring solutions are moved in the first subset. The second subset consists of solutions which are sampled from outside the current population according to specified strategies.

- This process is iterated until a suitable solution is found or the computing time expires.

We shall develop mathematical foundations for these procedures involving the formalism of Markov chains, random perturbations of dynamical systems, and statistical mechanics. Our main stress will be on the absence of the so-called *premature convergence* and on the parameters assessment which guarantees it. Obviously, the previous procedure shares some properties with the genetic algorithm (GA). However, the GA processes by sequentially applying a mutation operator on the population and then a crossover operator and finally a selection operator so that each step is composed of three basic operations [5, 10]. This amount of complexity creates an obstacle to the understanding of the process and to the correct assessment of its parameters. An important feature of the new procedure described is that the population is transformed from a single operation at each generation. This property greatly simplifies the subsequent Markov chain analysis.

2. The Markov chain approach

In this section, we formally describe the algorithms that are dealt with. For sake of simplicity, we assume that f is one-to-one on $f(E)$ and admits a unique minimal point that we denote by a^*. Since E is a discrete set and the focus is on the minimal point and not on the minimal value, there is no loss of generality in such an assumption. The principle of random search is to create a noisy dynamics on a set X (X may be different from E) for which the final state will correspond to the global minimum on E (as randomness vanishes). Our search procedure uses a population of n solutions which evolve simultaneously on E. Using notations from [5, 6], we set

$$X = E^n, \qquad n \geq 2. \tag{2.1}$$

For all $x = (x_1, \ldots, x_n) \in X$, we denote

$$[x] = \{x_1, \ldots, x_n\}. \tag{2.2}$$

Let x_* be the element for which f is minimal over $[x] \subset E$, that is, $x_* = \operatorname{argmin}_{x_i \in [x]} f(x_i)$.

We identify the uniform population $(a, \ldots, a)$ and the element $a \in E$ by denoting $(a) = (a, \ldots, a)$. To describe our algorithms, we need to introduce some parameters. The first parameter is $\epsilon > 0$, which will control the intensity of randomness. As far as convergence results are concerned, this parameter must go to zero. Its role is the same as *temperature* in the simulated annealing procedure [1, 3, 15]. Its role is to quantify the intensity of random changes in the population. The lower it is, the more changes are allowed. Let X_t^ϵ be the vector of trial solutions obtained at time t and suppose that $X_t^\epsilon = x \in X$. The basic algorithm is as follows.

Algorithm 1.

1. Build a subset $I \subset \{1, \ldots, n\}$ by putting i independently in I with a probability which is equal to
$$p^{\epsilon}_{\text{mut}} = \exp(-1/\epsilon). \tag{2.3}$$
2. If $i \in I$, choose $y_i \in E\backslash[x]$ with uniform probability.
3. If $i \notin I$, $y_i = x_*$.
4. Do $X^{\epsilon}_{t+1} = y$.

Building on the same structure, we can define many variants of this algorithm just by modifying either item 1 or item 2. Regarding item 2, an obvious possibility is to use a sampling procedure which is solution dependent (y_i depends on x and i). This hypothesis is not considered in this paper. As a variation on item 2, we instead consider a gradient-based procedure as follows.

Algorithm 2.

2′. If $i \in I$, choose $y_i \in E\backslash[x]$ with uniform probability. Compute
$$\Delta = f(y_i) - f(x_i). \tag{2.4}$$
If $\Delta < 0$ then accept y_i otherwise accept y_i with probability $\exp(-\Delta/\epsilon)$. Repeat the process until a y_i is actually accepted.

We also consider another variation of Algorithm 1 obtained by modifying the mutation probability. Modifying item 1 would permit the integration of nonuniform changes. Including more competitors when the population is far from the solution might speed up the optimization process. To do so, we introduce a second parameter $\lambda > 0$. We assume that

$$\lambda > f_{\max} \tag{2.5}$$

where $f_{\max}$ is the maximum value of f. The variant is as follows.

Algorithm 3.

1′. Build a subset $I \subset \{1, \ldots, n\}$ by putting i independently in I with a probability which is equal to
$$p_{\text{mut}} = \exp(-(\lambda - f(x_*))/\epsilon). \tag{2.6}$$

Equation (2.5) suggests that the cost function f must be rescaled so that large costs are less influential. However, we must warn of the danger of rescaling too much. Randomness would play a too important part and the population would wander as a random walk.

In the three algorithms, the random variable X_t^ϵ evolves as a Markov chain. We denote by Q_ϵ^i $(i = 1, 2, 3)$ the corresponding matrix of transition probabilities. We have the following logarithmic equivalents

$$\forall x, y \in X, \quad q_\epsilon^i(x, y) \sim \exp(-C_i(x, y)/\epsilon) \quad \text{as } \epsilon \to 0 \quad (i = 1, 2, 3) \tag{2.7}$$

with, for Algorithm 1,

$$C_1(x, y) = |I(x, y)|, \tag{2.8}$$

for Algorithm 2,

$$C_2(x, y) = |I(x, y)| + \sum_{i \in I(x,y)} (f(y_i) - f(x_i))^+, \tag{2.9}$$

and, for Algorithm 3,

$$C_3(x, y) = (\lambda - f(x_*))|I(x, y)|, \tag{2.10}$$

where $I(x, y)$ is the subset of $\{1, \ldots, n\}$ for which $y_i \notin [x]$ and we denoted $(\alpha)^+ = \max\{0, \alpha\}$.

The principle of these procedures is to keep a strong memory of the best current solution throughout the selection operation and evolve the population by introducing reasonable changes. The replacement of some solutions by new competitive ones is performed by sampling over the search space. As ϵ goes to zero, no more change occurs and selection acts as a deterministic mechanism on the whole population, leading to a uniform population in a single generation.

3. The statistical mechanics perspective

The basic idea underlying the convergence of algorithms 1, 2, and 3 is that a deterministic mechanism is perturbed at each step. This deterministic mechanism is easy to identify. It consists of assigning to each pool of solutions x the uniform element (x_*) (see also [5, 6]). In all cases, the Markov transition matrix Q_ϵ which is associated to the algorithm satisfies the classical convergence conditions of the Perron–Frobenius theorem. Thus, the Markov chain (X_t^ϵ) converges to a unique stationary distribution as t goes to infinity. The convergence of the algorithms relies upon the concentration of the stationary distribution on a^* as ϵ goes to zero. In [9], a theoretical framework is developed which is appropriate for dealing with the markovian perturbations of dynamical systems. We apply the results in [9] to simulated evolution. We give here a brief account of the Freidlin–Wentzell theory, adapting it to our specific context. Of course, a thorough exposition can be found in [9] (see also [6, 12, 13]). Consider a finite set X and the dynamical system defined by

$$\forall t \geq 0, \quad x_{t+1} = F(x_t), \quad x_0 \in X \tag{3.11}$$

with F a discrete map from X to itself. A markovian perturbation of the dynamical system (3.11) is a Markov chain (X_t^ϵ) on X such that the following logarithmic equivalent holds

$$\forall x, y \in X, \quad \mathrm{P}(X_{t+1}^\epsilon = y \mid X_t^\epsilon = x) \sim \exp(-\epsilon^{-1} C(x, y)) \quad \text{as } \epsilon \to 0. \tag{3.12}$$

The quantity $C(x, y)$ is called *one-step communication cost* between x and y. It expresses the difficulty for the chain (X_t^ϵ) to move from x to y in one step. This cost is presumed to satisfy the following conditions.

1. For all $x \in X$,

$$C(x, F(x)) = 0. \tag{3.13}$$

2. For all $x \in X$, $y \neq F(x)$,

$$C(x, y) > 0. \tag{3.14}$$

3. For all $(x, y) \in X \times X$, there exists a sequence $(x_0 \to x_1 \to \cdots \to x_r)$ such that

$$x_0 = x, \;\; x_i \in X, \;\; x_r = y, \quad \text{and} \quad \sum_{k=0}^{r-1} C(x_k, x_{k+1}) < \infty. \tag{3.15}$$

Every pair $(x, y) \in X \times X$ is weighted by the communication costs. Condition 1 formulates that no cost is allowed to the paths of the deterministic mechanism (3.11). By condition 3, every point $y \in X$ will be reachable from x in finitely many perturbative steps. These conditions warrant that the chain (X_t^ϵ) is ergodic and that its paths are close to that of the deterministic system as ϵ goes to zero. We assume that the set $\mathcal{A}$ of attractors of the dynamical system (3.11) consists of fixed points of f and denote $\mathcal{A} = \{a_1, \ldots, a_L\} \subset X$. The stationary distribution of the perturbed Markov chain concentrates around these fixed points. However, some of these points are favored with regard to the others. In [9], it is shown that the limiting perturbed dynamics (as $\epsilon \to 0$) can be described by introducing a Markov chain on the set of attractors of the dynamical system. The *communication cost* from a_i to a_j in $\mathcal{A}$ is defined in [9] as

$$V(a_i, a_j) = \min \left\{ \sum_{t=0}^{r-1} C(x_t, x_{t+1}), \; x_0 = a_i, \; x_t \in X, \; x_r = a_j, \; r \geq 1 \right\}. \tag{3.16}$$

The *virtual energy* function is defined on the set $\mathcal{A}$ by:

$$\forall l = 1, \ldots, L, \quad W(a_l) = \min_{g \in \mathcal{G}_{\mathcal{A}}(a_l)} \sum_{(a_i \to a_j) \in g} V(a_i, a_j). \tag{3.17}$$

In equation (3.17), the minimum is taken on the set of all a_l-graphs on $\mathcal{A}$ and the sum runs over the edges of these graphs. Recall that an x-graph ($g \in \mathcal{G}(x)$) ends at x and contains no cycle (each $y \neq x$ is the starting point of exactly one edge). The virtual energy W describes the asymptotic behavior of the chain (X_t^ϵ) as the perturbation vanishes. In [9], a logarithmic equivalent of the (stationary) probability that the perturbed process visits the attractor a_l is obtained:

$$\forall l = 1, \ldots, L, \quad p_l^\epsilon \sim \exp\left(\frac{W_{\min} - W(a_l)}{\epsilon}\right) \quad \text{as } \epsilon \to 0. \tag{3.18}$$

Let $\mathcal{W}^*$ be the set of points in $\mathcal{A}$ for which the minimum $W_{\min}$ of W is attained. Equation (3.18) states that the distribution concentrates on $\mathcal{W}^*$ (see also [6, 13]). As $\epsilon \to 0$, the stationary distribution behaves as a gibbsian distribution associated with the energy W. The most probable states are the states for which $W(a_l)$ is low. As far as stochastic optimization procedures are concerned, the statistical mechanics perspective is natural. This was the point of view of simulated annealing based procedures. However, the energy function in these procedures was always chosen equal or closely related to the cost function f itself. In our approach, the function W might be very different from f. In such a case there is no guarantee that convergence holds. Additional conditions on the communication costs ensure the concentration of the stationary distribution around the best solution a^*.

Theorem 3.1. *Assume that there exists an $a^* \in \mathcal{A}$ such that*

$$\forall x, y \in \mathcal{A},\ y \neq a^* \ , \quad V(x, a^*) < V(a^*, y). \tag{3.19}$$

Then, for all $x \neq a^*$, $W(a^*) < W(x)$ (the chain concentrates on a^*).

We turn now to Algorithms 1, 2, and 3. We shall prove that the previous result ensures that premature convergence is avoided and that the Markov chain (X_t^ϵ) concentrates on (a^*) in the long run. First of all, we must check that the quantities $C(x, y)$ define one step communication costs for all algorithms. Actually, the sole condition to check is item 2. We must have

$$C_i(x, y) > 0 \quad \forall y \neq (x_*) \quad i = 1, 2, 3 \tag{3.20}$$

which is immediate in Algorithms 1 and 2 and results in

$$\lambda > f_{\ \max} \tag{3.21}$$

in Algorithm 3.

Theorem 3.2. (Concentration on a^*). *Consider the Markov chains defined by Algorithms 1 and 2. Then, we have*

$$\mathcal{W}^* = \{(a^*)\} \tag{3.22}$$

with a^* the unique minimum of f. If $\lambda > f_{\ \max}$, the same conclusion holds for Algorithm 3.

Proof. Let $a \in E$, $a \neq a^*$. We first deal with Algorithm 1. We have

$$V((a),(a^*)) = C_1((a),(a^*,a,\dots,a)) = 1 \tag{3.23}$$

and

$$V((a^*),(a)) = C_1((a^*),(a)) = n. \tag{3.24}$$

Now we can apply Theorem 3.1 to reach the conclusion. With regard to Algorithm 2, we must have

$$n > \frac{1}{1+\delta} \tag{3.25}$$

with $\delta = f(a) - f_{\min}$. This condition is obviously satisfied. Finally, concerning Algorithm 3, the condition to check is

$$n > \frac{\lambda - f(a)}{\lambda - f_{\min}} \tag{3.26}$$

which is again obviously satisfied. ■

Comments

It is possible to combine items $1'$, $2'$, and 3 to obtain a fourth algorithm. The same conclusion holds for this algorithm as far as condition (3.21) is satisfied.

4. Conclusion

New simulated evolution algorithms are presented for global minimization. The main feature of these algorithms is to couple the standard mutation and selection operators into a single one. Exponential probabilities were also introduced so as to stress the relevance of statistical mechanics arguments. The work followed the stream of techniques developed to analyze the simulated annealing [1, 4, 13] and genetic algorithms [5, 6] by discrete versions of the Freidlin–Wentzell theory. Many computer simulations of the four algorithms have been performed on quadratic test functions. The best performances were obtained with Algorithms 2 and 4. Moreover, an additional (significant) gaussian noise was added to those functions and good convergence properties were also observed. These algorithms were also run on knapsack problems with again the best results for Algorithm 2. However, the performances depend on a good choice of the parameter ϵ. This parameter must be large enough to allow convergence in a reasonable computing time. We are aware that our study must be completed by the comparison of the algorithms. Unfortunately, the concentration is only an asymptotical property. In practice, ϵ never goes to zero and it does not make sense to compare the different algorithms at the same value of ϵ. On the other hand, a theoretical study would involve asymptotics on the spectrum of the corresponding Markov chains. Such results are available [9] but deserve a specific study. We emphasize that concentration is the main point in practical situations.

Acknowledgments

I wish to thank Dr. Daniela Zaharie for fruitful conversations.

Appendix

We give a proof of Theorem 3.1. Let $x \neq a^*$ with g being a graph for which

$$W(x) = \sum_{(a_i \to a_j) \in g} V(a_i, a_j).$$

We built an a^*-graph by deleting the edge $(a^* \to y)$ in g and introducing the edge $(x \to a^*)$. Thus, we have

$$W(a^*) \leq W(x) + V(x, a^*) - V(a^*, y) < W(x). \blacksquare$$

References

[1] R. Azencott, "Simulated Annealing," *Astérisque, Séminaire Bourbaki*, **697** (1988) 161–175.

[2] T. Bäck, *Evolutionary Algorithms in Theory and Practice* (Oxford University Press, USA, 1996).

[3] D. Bertsimas and J. Tsitsiklis, "Simulated Annealing," *Statistical Science*, **8** (1993) 10–15.

[4] O. Catoni, "Rough Large Deviations Estimates for Simulated Annealing. Application to Exponential Schedules," *Annals of Probability*, **20** (1992), 1109–1146.

[5] R. Cerf, "Asymptotic Convergence of a Genetic Algorithm," *Comptes Rendus de l'Académie des Sciences Paris, Série I*, **319** (1994) 271–276.

[6] R. Cerf, "The Dynamics of Mutation Selection Algorithms with Large Population Sizes," *Annales de l'Institut Henri Poincaré Probabilité et Statistique*, **32** (1996) 455–508.

[7] D. B. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence* (IEEE Press, New-York, 1995).

[8] O. François and D. Zaharie, "Markovian Perturbations of Discrete Iterations: Lyapunov Functions, Global Minimization and Associative Memory," preprint, Grenoble, 1997.

[9] M. I. Freidlin and A. D. Wentzell, *Random Perturbations of Dynamical Systems* (Springer-Verlag, New York, 1984).

[10] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning* (Addison Wesley, Reading, MA, 1989).

[11] J. H. Holland, *Adaptation in Natural and Artificial Systems* (University of Michigan Press, Ann Arbor, 1975).

[12] C. R. Hwang and S. J. Sheu, "Singular Perturbed Markov Chains and Exact Behaviour of the Simulated Annealing Process," *Journal of Theoretical Probability*, **5** (1992) 223–249.

[13] A. Trouvé, "Cycle Decomposition and Simulated Annealing," *SIAM Journal on Control and Optimization*, **34** (1996) 966–986.

[14] J. N. Tsitsiklis, "Markov Chains with Rare Transitions and Simulated Annealing," *Mathematics of Operation Research*, **14** (1989) 70–90.

[15] P. J. M. Van Laarhoven and E. H. L Aarts, *Simulated Annealing: Theory and Applications* (Reidel, Dordrecht, 1987).