The Factorized Distribution Algorithm for Additively Decomposed Functions

Heinz Mühlenbein
Real World Computing Partnership
Theoretical Foundation GMD Laboratory
53754 Sankt Augustin, Germany
muehlenbein@gmd.de

Thilo Mahnig
Real World Computing Partnership
Theoretical Foundation GMD Laboratory
53754 Sankt Augustin, Germany
Thilo.Mahnig@gmd.de

Published in: *Proceedings of the 1999 Congress on Evolutionary Computation*, 1999, IEEE Press, 752–759

Abstract- FDA - the Factorized Distribution Algorithm - is an evolutionary algorithm that combines mutation and recombination by using a distribution. First the distribution is estimated from a set of selected points. It is then used to generate new points for the next generation. In general a distribution defined for n binary variables has 2^n parameters. Therefore it is too expensive to compute. For additively decomposed discrete functions (ADFs) there exists an algorithm that factors the distribution into conditional and marginal distributions, each of which can be computed in polynomial time. The scaling of FDA is investigated theoretically and numerically. The scaling depends on the ADF structure and the specific assignment of function values. Difficult functions on a chain or a tree structure are optimized in about $O(n\sqrt{n})$ function evaluations. More standard genetic algorithms are not able to optimize these functions. FDA is not restricted to exact factorizations. It also works for approximate factorizations.

Keywords – evolutionary algorithms, graphical models, factorization of distributions, Boltzmann selection

1 Introduction

It is well known that evolutionary algorithms have difficulties in optimizing functions with nonlinear interacting variables. For continuous variables Rosenbrock's function is a classical example. In order to optimize this function search points have to be generated in a small valley running nonorthogonal to all axes. All variables have to be changed together in a certain manner in order to obtain an improvement. In this paper we investigate the problem of nonlinear interacting variables for additively decomposed functions (ADFs) defined on discrete domains.

A number of new evolutionary algorithms have been proposed which optimize ADFs better than genetic algorithms. These algorithms try to detect and exploit the structure of an ADF. The methods used can be classified as follows:

- Adaptive recombination
- Explicit detection of relations [7]
- Dependency trees [3]
- Bivariate marginal distributions [17]
- Estimation of distributions [16],[4]

Adaptive recombination uses a number of heuristics to modify two-parent recombination. Kargupta's GEMGA [8] tries to detect dependency relations by manipulating individual substrings.

The last three methods are based on probability theory and statistics. They use all the statistical information contained in the population of selected points to detect dependencies. In this paper an algorithm called the Factorized Distribution Algorithm (FDA) will be investigated. FDA uses an exact factorization of the distribution of selected points.

FDA is based on a solid mathematical foundation. Many results can be derived by mathematical analysis. Therefore this paper is a mixture between theoretical analysis and numerical experiments. The experiments are mainly used to confirm the theoretical analysis.

The outline of the paper is as follows. In Section 2 the main factorization theorem is recalled. FDA is defined in 3. We make a theoretical analysis of FDA for large (infinite) populations in Section 6. The extension of FDA to continuous variables is briefly discussed.

2 A Factorization Theorem

In this section we recall the main results proven in [14]. First we define precisely an ADF.

Definition: An *additively decomposed function* (ADF) is defined by

$$f(x) = \sum_{s_i \in S} f_i(\Pi_{s_i} x) \qquad S = \{s_1, \ldots, s_l\} \quad s_i \subseteq \tilde{X} \quad (1)$$

where

$$\begin{aligned}
\tilde{X} &:= \{x_1, \ldots, x_n\} \quad \mathbf{B} := \{0,1\} \quad X := \mathbf{B}^{|\tilde{X}|} \\
X_s &\subseteq X \text{ with } s \subseteq \tilde{X} \\
\Pi_s x &:= \text{the projection of } x \in X \text{ onto the subspace } X_s
\end{aligned}$$

Next we define a distribution which will be used for generating promising points. A good candidate is a generalization of the Gibbs or Boltzmann distribution.

Definition: *The Gibbs or Boltzmann distribution of a function f is defined for $u \geq 1$ by*

$$p(x) := \frac{\mathrm{Exp}_u f(x)}{\sum_y \mathrm{Exp}_u f(y)} \tag{2}$$

where for notational convenience

$$\mathrm{Exp}_u f(x) := u^{f(x)} \quad F_u := \sum_y \mathrm{Exp}_u f(y)$$

Remark: The Boltzmann distribution is usually defined as $e^{-\frac{g(x)}{T}}/Z$. The term $g(x)$ is called the energy. Setting $g(x) = 1/f(x)$ and $u = e^{-\frac{1}{T}}$ gives Equation 2. $Z = F_u$ is called the *partition function.*

The Boltzmann distribution has the following feature: the larger the function value $f(x)$ becomes, the larger $p(x)$ becomes (for $u > 1$). It seems a good optimization strategy to distribute the search points in such a manner. Unfortunately the computation of the Boltzmann distribution needs an exponential effort (in the size of the problem). There are at least two approaches to reduce the computation: to approximate the Boltzmann distribution or to look for ADFs where the distribution can be computed in polynomial time. The first approach is used by *Simulated Annealing* [1]. FDA is based on the second approach. The distribution is factored into a product of marginal and conditional probabilities. They are defined as usual

$$\begin{aligned} p(\Pi_{c_i} x) &= \sum_{y \in X, \Pi_{c_i} y = \Pi_{c_i} x} p(y) && (3) \\ p(\Pi_{b_i} x | \Pi_{c_i} x) &= \frac{p(\Pi_{b_i} x, \Pi_{c_i} x)}{p(\Pi_{c_i} x)} && (4) \end{aligned}$$

The main factorization theorem uses the following sequence of sets as input.

Definition: *Given a set of sets $S = \{s_1, \ldots, s_l\}$, we define for $i = 1, 2, \ldots, l$ sets d_i,b_i and c_i*

$$\begin{aligned} d_i &:= \textstyle\bigcup_{j=1}^{i} s_j && (5) \\ b_i &:= s_i \setminus d_{i-1} && (6) \\ c_i &:= s_i \cap d_{i-1} && (7) \end{aligned}$$

We set $d_0 = \emptyset$.
In the theory of decomposable graphs, d_i are called *histories*, b_i *residuals* and c_i *separators* [10].

Theorem 1 (Factorization Theorem) *Let $p(x)$ be a Boltzmann distribution on X with*

$$p(x) = \frac{Exp_u f(x)}{F_u} \quad \text{with } u > 1 \text{ arbitrarily.} \tag{8}$$

If

$$\begin{aligned} b_i \neq \emptyset \quad \forall i = 1, \ldots, l; \quad d_l = \tilde{X}, && (9) \\ \forall i \geq 2 \; \exists j < i \;\; \text{such that } c_i \subseteq s_j && (10) \end{aligned}$$

then

$$p(x) = \prod_{i=1}^{l} p(\Pi_{b_i} x | \Pi_{c_i} x) \tag{11}$$

The proof can be found in [14]. There also the design rational of the Factorized Distribution Algorithm **FDA** and its connection to genetic algorithms is discussed. Equation 10 is called the *running intersection property*.

The running intersection property is fulfilled if the interaction graph derived from the sets s_i is similar to a tree [10].

Therefore the class of ADFs with a numerical efficient exact factorization is limited, as the following 2-D Ising spin systems shows.

$$F_{Ising}(y) = \sum_i \sum_{j \in N(i)} J_{ij} x_i x_j \quad x_i \in \{-1, 1\} \tag{12}$$

The sum is taken over the four spatial neighbors $N(i)$, but each J_{ij} is used only once. The objective function is purely quadratic. All factorizations fulfilling the running intersection property on 2-D grids need large sets. We state without proof.

Proposition 2: *All exact factorizations of ADFs defined on 2-D grids require the computation of conditional marginal distributions of size $O(\sqrt{n})$ where n is the size of the grid.*

3 The Factorized Distribution Algorithm

We assume that an ADF and a factorization of the probability distribution is given. The factorization can also be used at the initialization. For faster convergence a proportion of $r * N$ individuals can be generated with a local approximation of the conditional marginal distributions. The local approximation is explained in [14].

$\mathbf{FDA_r}$

- **STEP 0:** Set $t \Leftarrow 0$. Generate $(1 - r) * N \gg 0$ points randomly and $r * N$ points according to the local approximation.
- **STEP 1:** Selection of promising points.
- **STEP 2:** Compute the conditional probabilities $p^s(\Pi_{b_i} x | \Pi_{c_i} x, t)$.
- **STEP 3:** Generate a new population according to $p(x, t + 1) = \prod_{i=1}^{l} p^s(\Pi_{b_i} x | \Pi_{c_i} x, t)$.
- **STEP 4:** If the termination criteria are met, FINISH.

- **STEP 5:** Add the best point of the previous generation to the generated points (elitist).
- **STEP 6:** Set $t \Leftarrow t+1$. Go to STEP 2.

FDA can be used with an exact or an approximate factorization. It uses *finite samples* of points. Convergence of FDA to the optimum will depend on the size of the samples. FDA can be run with any popular selection method. We usually apply truncation selection. A comparison between Boltzmann selection and truncation selection is made in Section 6.

3.1 Analysis of Factorization

The computational complexity of FDA depends on the factorization and the population size N. The number of function evaluations to obtain a solution is given by

$$FE = GEN_e * N \tag{13}$$

GEN_e denotes the number of generations till convergence. Convergence means that $p(x,t+1) = p(x,t)$. The computational complexity of computing N new search points is given by

$$compl(Npoints) \approx l * N \tag{14}$$

$|s_i|$ denotes the number of elements in set s_i. The computational complexity of computing the probability is given by

$$compl(p) \approx (\sum_{i=1}^{l} 2^{|s_i|}) * M \tag{15}$$

where M denotes the number of selected points. We therefore obtain that the amount of computation of FDA depends on N and the size of the defining sets s_i. In order to exactly compute the probabilities an infinite population is needed. But a numerical efficient *FDA* should use a minimal population size N^* still giving good numerical results. The computation of N^* is a difficult problem for any search method using a population of points. This problem will be discussed in Section 7.

FDA furthermore depends on the defining sets s_i. We have implemented a simple factorization algorithm which assumes that the defining sets are sorted into a sequence $(s_1, s_2, \ldots, s_n)$. Then the sets b_i and c_i such that $b_i \neq \emptyset$ are computed according to the factorization theorem. Changing the sequence will change the factorization. For the root set b_1 the sub function which is maximally nonlinear (measured as deviance from a linear square predictor) is chosen.

Computing a factorization with minimal complexity for an arbitrary ADF is a very difficult task. We conjecture that this problem is in NP. This research needs deep results from graph theory. The problem of factorization of a probability distribution is also dealt with in the theory of *graphical models* [6]. Any progress in the theory of graphical models can also be used for FDA.

4 Convergence of FDA

Mühlenbein et al. [14] proved convergence of FDA if points are selected according to a Boltzmann distribution with a given $v > 1$. In this case the distribution p^s of the selected points is given by

$$p^s(x,t) = p(x,t)\frac{\mathrm{Exp}_v f(x)}{\sum_x p(x,t) Exp_v f(x)} \quad . \tag{16}$$

Boltzmann selection has been investigated for genetic algorithms by de la Maza & Tidor [5]. One can easily show that if $p(x,t)$ is a Boltzmann distribution, then $p^s(x,t)$ is also a Boltzmann distribution. If new points are generated according to

$$p(x,t+1) = p^s(x,t),$$

then $p(x,t+1)$ obviously is a Boltzmann distribution. The following two theorems have been proven in [14].

Theorem 2 *If the initial points are distributed according to* $p(x,0) = \frac{Exp_u f(x)}{F_u}$ *with* $u \geq 1$*, then for FDA the distribution at generation t is given by*

$$p(x,t) = \frac{Exp_w f(x)}{\sum_y Exp_w f(y)} \tag{17}$$

with $w = u \cdot v^t$*.*
From this theorem convergence easily follows.

Theorem 3 *Let* $X_{opt} = \{x_{1opt}, x_{2opt}, ..\}$ *be the set of optima. Then under the assumptions of Theorem 2*

$$\lim_{t\to\infty} p(x,t) = \begin{cases} \frac{1}{|X_{opt}|} & x \in X_{opt} \\ 0 & else \end{cases} \tag{18}$$

Therefore FDA with Boltzmann selection has a solid theoretical foundation. Unfortunately Boltzmann selection has numerical drawbacks. This will be explained in the next Section 6. We mainly use FDA with *truncation selection*. It works as follows. Given is a truncation threshold τ. The best $\tau * N$ individuals are selected. We estimate the conditional probabilities of the selected points $p^s(\Pi_{b_i} x | \Pi_{c_i} x, t)$ from the empirical distribution. Then the factorization theorem is used to generate new search points according to

$$p(x,t+1) = \prod_{i=1}^{l} p^s(\Pi_{b_i} x | \Pi_{c_i} x, t)$$

5 Continuous Variables

Interacting variables also pose a problem for all continuous optimization methods. For simplicity we assume that x_i is restricted to an interval $[a_i, b_i]$. Then the domain considered for optimization is $D := [a,b]^n$. The theory presented can also be applied to continuous variables with minor modifications only.

Definition: *The continuous Gibbs or Boltzmann distribution of a function f is defined for $u \geq 1$ by*

$$p(x) := \frac{\mathrm{Exp}_u f(x)}{\int_D \mathrm{Exp}_u f(y) dy} \tag{19}$$

$Z_u := \int_D \mathrm{Exp}_u f(y) dy$ is called the partition function. This shows the minor modification necessary. Sums have to be changed into integrals. Similarly we define Boltzmann selection for FDA at generation t for basis $v > 1$

$$p^s(x,t) = p(x,t) \frac{\mathrm{Exp}_v f(x)}{\int_D p(y,t) Exp_v f(y) dy} \quad . \tag{20}$$

We easily obtain a theorem similar to the one for discrete variables.

Theorem 4 *If the initial points are distributed according to $p(x,0) = \frac{Exp_u f(x)}{Z}$ with $u \geq 1$, then for FDA the distribution at generation t is given by*

$$p(x,t) = \frac{Exp_w f(x)}{\int_y Exp_w f(y) dy} \tag{21}$$

with $w = u \cdot v^t$.
Proof: We just prove one step, from the initial generation $t = 0$ to the first generation $t = 1$. We have

$$\begin{aligned} p(x,1) &= \frac{\mathrm{Exp}_u f(x)}{Z_u} \frac{\mathrm{Exp}_v f(x)}{\int_D \frac{Exp_u f(x)}{Z} \mathrm{Exp}_v f(x) dx} \\ &= \frac{Exp_{u*v} f(x)}{\int_D Exp_{u*v} f(x) dx} \end{aligned}$$

Repeated application of this equation gives the assertion 21. ■

This proof is almost identical to the proof for discrete variables. Thus the main factorization theorem holds also for continuous variables.

Theorem 5 (Factorization Theorem) *Let $p(x)$ be a continuous Boltzmann distribution on D with*

$$p(x) = \frac{Exp_u f(x)}{Z_u} \quad \textit{with } u > 1 \textit{ arbitrarily.} \tag{22}$$

If

$$b_i \neq \emptyset \quad \forall i = 1, \ldots, l; \quad d_l = \tilde{X}, \tag{23}$$

$$\forall i \geq 2 \, \exists j < i \;\; \textit{such that } c_i \subseteq s_j \tag{24}$$

then

$$p(x) = \prod_{i=1}^{l} p(\Pi_{b_i} x | \Pi_{c_i} x) \tag{25}$$

The discrete conditional distributions $p^s(\Pi_{b_i} x | \Pi_{c_i} x)$ can be easily computed from empirical data. But for continuous variables the computation of the marginal probabilities requires integration over subspaces of D. This computation is numerically too expansive.

Therefore additional assumptions concerning the distributions have to be made. We discuss the problem with an example. Let the fitness function be defined by

$$f(x) = -\frac{1}{2}(x-\mu)^T B (x-\mu) \tag{26}$$

where B is a symmetric, positive definite matrix. The domain D has to be sufficiently large (see below). Using the Boltzmann distribution, we can use the main factorization theorem to obtain the subsets s_i and a corresponding factorization.

For this example the Boltzmann distribution is just a multivariate normal distribution scaled by the temperature T. The parameters of this distribution can be computed as usual from the mean μ and the variance/covariance matrix A. Then $A = B^{-1}$ [9]. If the running intersection property is fulfilled, then the distribution factorizes. Because it is a multivariate normal distribution, samples can be fitted by estimating means, variances and covariances (setting covariances to zero according to the original interaction graph). The domain D has to be large enough so that no significant portion of the multivariate normal distribution is cut off *a priori*.

When the temperature increases, the matrix of variances and covariances is scaled so that the distribution is more and more concentrated around the peak at μ.

If, for example, B is tridiagonal ($b_{ij} = 0$ if $|i-j| > 1$), then so is A, and f has the following structure:

$$\begin{aligned} f(x) &= -\frac{1}{2}(x_1-\mu_1)^2 b_{11} + \sum_{i=2}^{n} \Big[-\frac{1}{2}(x_i-\mu_i)^2 b_{ii} \\ &\qquad + (x_{i-1}-\mu_{i-1})(x_i-\mu_i) b_{i-1,i} \Big] \\ &= f_1(x_1) + \sum_{i=2}^{n} f_i(x_{i-1}, x_i) \end{aligned}$$

The distribution factorizes as a chain:

$$p(x) = p(x_1) p(x_2|x_1) p(x_3|x_2) \ldots p(x_n|x_{n-1}).$$

Marginal probabilities of multivariate normal distributions are also multivariate normal [9]. The factorization enables us to estimate the parameters of these *local* multivariate normal distributions *locally*. Take as example

$$p(x_2|x_1) = \frac{p(x_1,x_2)}{p(x_1)}$$

Then $p(x_1,x_2)$ is a bivariate normal distribution and $p(x_1)$ is a normal distribution. The whole distribution can thus be described by $3n-1$ parameters (n means, n variances and $n-1$ covariances), that is, the same number of parameters as in the function definition.

Note that this method resembles evolution strategies and evolutionary programming [2]. All three methods generate new search points according to a multivariate normal distribution. FDA has a global view. It computes the normal distribution which gives the best fit to *all* selected points. The other two strategies plaace the mean of the normal distribution at the best point computed so far.

If the selected points are approximately distributed like a normal distribution, then FDA will converge fast to the optimum. But if the empirical distribution is multimodal, a fit by a normal distribution would be bad. Therefore FDA has to use more general methods for fitting continuous distributions in order to optimize arbitrary ADFs. This topic is under investigation.

6 Analysis of FDA for Large Populations

For Boltzmann selection we have analytically derived exact difference equations for the marginal distributions. FDA mainly depends on the factorization, not on the *function values*. Numerical experiments have confirmed that the behaviour of FDA is very similar for functions having a similar factorization.

Typical fitness distributions are generated by the two functions

$$OneMax(n) = \sum_{i=1}^{n} x_i \tag{27}$$

$$Int(n) = \sum_{i=1}^{n} 2^{i-1} x_i \tag{28}$$

$OneMax$ has $(n+1)$ different different fitness values which are binomial or multinomial distributed. Int has 2^n different fitness values. For ADFs the multinomial distribution is “typical”, i.e it occurs fairly often. The distribution generated by Int is more special. Both functions are linear and therefore the following factorization is used

$$p(x, t+1) = \prod_{i=1}^{n} p(x_i, t) \tag{29}$$

We first analyze $OneMax$.

Theorem 6 *Select points according to a Boltzmann selection with basis $v > 1$. Then the distribution generated by FDA for $OneMax$ is given by*

$$p(x,t) = \frac{v^{tf(x)}}{(1+v^t)^n} \tag{30}$$

The number of generations needed to generate the optimum with probability $1-\epsilon$ is given by

$$GEN_\epsilon \approx \frac{ln\frac{n}{\epsilon}}{ln(v)} \tag{31}$$

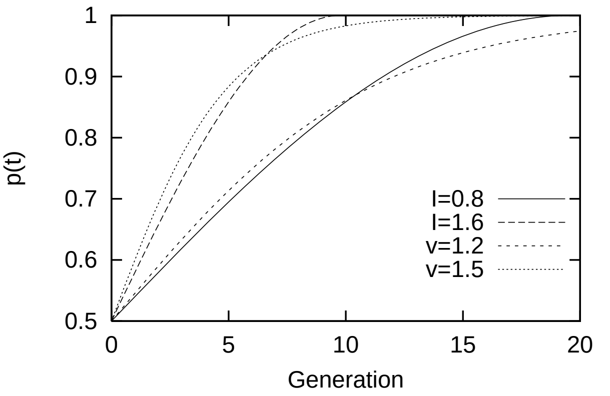

Figure 1: Probability $p(t)$ for $OneMax(100)$ with Truncation selection and Boltzmann selection

For truncation selection an approximate analysis was already done in [13],[11]. For simplicity we assume that in the initial population all univariate marginal distributions are equal ($p_i(x_i = 1, t = 0) = p_0$). Then $p_i(x_i = 1, t) := p(t)$ for all t.

Theorem 7 *For truncation selection τ with selection intensity I_τ the marginal probability $p(t)$ obeys for $OneMax$*

$$p(t+1) = p(t) + \frac{I_\tau}{n}\sqrt{np(t)(1-p(t))}. \tag{32}$$

This equation has the approximate solution ($p_0 := p(0)$)

$$p(t) = 0.5\left(1 + sin\left(\frac{I_\tau}{\sqrt{n}}t + arcsin(2p_0 - 1)\right)\right) \tag{33}$$

where

$$t \le \left(\frac{\pi}{2} - arcsin(2p_0 - 1)\right)\frac{\sqrt{n}}{I_\tau}$$

The number of generations till convergence is given by

$$GEN_e = \left(\frac{\pi}{2} - arcsin(2p_0 - 1)\right)\frac{\sqrt{n}}{I_\tau}. \tag{34}$$

The relation between τ and I_τ depends on the fitness distribution [11]. Assuming that the fitness distribution is normal, I_τ can be computed from the error integral. For normal distribution we have

$$I_\tau \approx \alpha k \qquad \text{for} \qquad \tau = 2^{-k} \quad k \ge 2$$

Asymptotically truncation selection needs more generations to convergence than Boltzmann selection. GEN_e is of order $O(ln(n))$ for Boltzmann selection and of order $O(\sqrt{n})$ for truncation selection. But if the basis v is small (e.g. $v = 1.2$), and $\epsilon = 0.01$ then even for $n = 1000$ truncation selection converges faster than Boltzmann selection.

The different behaviour of Boltzmann selection and truncation selection is shown in Figure 1. The two equations

30 and 33 are plotted for reasonable values of v and I. For $v = 1.2$ Boltzmann selection selects slightly more severe than truncation selection with $I = 0.8$ at the beginning. Boltzmann selection gets weak when the population approaches the optimum. The same behaviour can be observed for $v = 1.5$. In fact, all selection methods using proportionate or exponential proportionate selection have this problem. If the fitness values in the population differ only slightly, selection gets weak. Truncation selection does not have this problem. It selects much stronger than Boltzmann selection when approaching the optimum. Therefore truncation selection with $I = 1.6$ converges faster than Boltzmann selection for $v = 1.5$.

The convergence of Boltzmann selection can be speeded up if an annealing schedule is used. This means that the basis v has to be changed during the run. The optimal schedule depends on the given fitness function. We will investigate this problem for the fitness distribution generated by $OneMax$. We will show that FDA with truncation selection generates a Boltzmann distribution. Therefore for each truncation threshold τ there exists a corresponding annealing schedule for Boltzmann selection generating the same distributions.

Theorem 8 *Let the distribution be generated by* $p(x) = \prod_{i=1}^n p(x_i)$. *Let* $p(x_i = 1) := p_i$. *Then there exist* $T_1, \ldots, T_n$ *with*

$$\frac{1}{T_i} = ln \frac{p_i}{1-p_i} \tag{35}$$

so that for $f(x) = \sum_{i=1}^n x_i$

$$p(x) = \frac{e^{\sum_{i=1}^n \frac{x_i}{T_i}}}{Z} \tag{36}$$

Z *is the partition function defined by* $\sum_y p(y) = 1$.

Proof: Because $exp\,(ln(p_i/(1-p_i))) = p_i/(1-p_i)$ one obtains

$$Z = 1 + \frac{p_1}{1-p_1} + \ldots + \frac{p_n}{1-p_n} + \frac{p_1 p_2}{(1-p_1)(1-p_2)} + \\ \cdots \frac{p_{n-1}p_n}{(1-p_{n-1})(1-p_n)} + \cdots \frac{\prod_i p_i}{\prod_i (1-p_i)}$$

This can be simplified to

$$Z = \frac{1}{\prod_{i=1}^n (1-p_i)}$$

Now the conjecture easily follows. ■

Corollary *If* $p_1 = \ldots = p_n := p$ *then* $p(x)$ *is a Boltzmann distribution with*

$$p(x) = \frac{e^{\frac{f(x)}{T}}}{Z} \tag{37}$$

where

$$\frac{1}{T} = ln \frac{p}{1-p} \tag{38}$$

For $p = 1/2$ we have $T = \infty$ and for $p = 1$ we get $T = 0$. Using Equation 32 we can now compute the annealing schedule. It is given by

$$\frac{1}{T(t)} = ln \frac{p(t)}{1-p(t)} \tag{39}$$

In Table 1 the schedule is shown for $n \in \{16, 32, 64, 256\}$. $1/T(t)$ first grows linearly in t, it increases nonlinear when approaching the optimum.

t	$n = 16$	$n = 32$	$n = 64$	$n = 256$
1	0.4055	0.2848	0.2006	0.1000
2	0.8377	0.5785	0.4044	0.2005
3	1.3238	0.8884	0.6135	0.3016
4	1.9125	1.2244	0.8305	0.4036
5	2.7213	1.6005	1.0587	0.5068
6	4.2842	2.0401	1.3020	0.6114
7	∞	2.5878	1.5658	0.7179
8		3.3510	1.8574	0.8265
9		4.7853	2.1880	0.9376
10		∞	2.5756	1.0517
11			3.0530	1.1693
12			3.6911	1.2907
13			4.7096	1.4168
14			∞	1.5482
28				5.6610
29				7.5458
30				∞

Table 1: Value of $1/T(t)$ for $OneMax$ and $\tau = 0.5$

Let us now turn to the analysis of the function Int. We first consider truncation selection with $\tau = 0.5$ and a large population size. After one generation of selection the n-th bit will be fixed. The other bits will not be affected by selection. After the next generation bit $(n-1)$ will be fixed etc. Convergence to the optimum is achieved after n generations.

For truncation selection with $\tau = 0.25$ two bits will be fixed in every generation. Convergence will be reached after $n/2$ generations. Therefore we obtain for Int.

Theorem 9 *For truncation selection with* $\tau = 2^{-k}; k \geq 1$ *the number of generations to converge to the optimum for* Int *is given by*

$$GEN_e = \frac{n}{k} \tag{40}$$

Setting the selection intensity $I_\tau = k$ for $\tau = 2^{-k}$ we obtain the same result as for $OneMax$: GEN_E scales inversely proportionate to I_τ. But the scaling is different for n. GEN_e scales proportionate to n. This is the worst case, as the following theorem shows:

Theorem 10 *Let the optimum be unique. If the population size is very large we have for truncation selection with $\tau = 2^{-k}$*

$$GEN_e \leq \frac{n}{k} \tag{41}$$

Proof: In an infinite population the optimum is contained with probability $1/2^n$ if it is unique. After one step of selection the probability will be increased at least to $2^k/2^n$. In about n/k steps the probability of the optimum has increased 1. ■

The theoretical analysis of Int for Boltzmann selection is more difficult and will be omitted.
We summarize the results for truncation selection.

- For $\tau_k = 2^{-k}$ $k \geq 2$ we have approximately $I_{\tau_k}/I_{\tau_{k+1}} = k/(k+1)$
- GEN_e is bounded by n/I_τ for $\tau \leq 1/2$
- For "typical" fitness distributions GEN_e is proportionate to $\sqrt{n}/I_\tau$ for $\tau \leq 1/2$.

This means that FDA will converge in at most n steps for $\tau \leq 0.5$. The difficult part remaining is the computation of an "optimal" τ. For this investigation we have to compute for each τ the critical population size $N^*(\tau)$.

7 Estimation of the Optimal Selection Intensity

A big problem of all population based search methods is their dependency on the population size. Here FDA with truncation selection has a nice numerical property. If the population size is larger than N^* then

$$GEN_e(N) = GEN_e(N = \infty) \quad N \geq N^* \tag{42}$$

This behaviour has been confirmed by many numerical experiments. It means that the number of generations to converge remains constant for $N \geq N^*$. N^* is called the *critical population size*, defined as the minimal population size needed to find the optimum with high probability, e.g. 99%. The determination of the critical population size N^* is difficult.

N^* obviously depends on the truncation threshold. The smaller the threshold τ, the larger $N^*(\tau)$. This has been first investigated by Mühlenbein and Schlierkamp-Voosen [12] for $OneMax$ and genetic algorithms. A more detailed investigation can be found in [13]. If $N^*(\tau)$ has been determined, then an optimal truncation threshold τ_{opt} can be computed. This threshold gives the minimum number of function evaluations FE.

Definition: The optimum truncation threshold τ_{opt} is defined by

$$\tau_{opt} = \min_\tau FE(\tau) = \min_\tau GEN_e(\tau) * N^*(\tau) \tag{43}$$

The following theorem has been derived from a Markov chain analysis. The Markov model is simplified, therefore we just conjecture.

Conjecture: *Let $\tau_k = 2^{-k}$. For FDA with fitness function Int the critical population size $N^*(\tau)$ is approximately given by*

$$N^*(\tau_k) \approx N^*(\tau_1) * 2^{\frac{k-1}{2}} \qquad k \geq 1$$

The following result follows from $k > 1$ from the above conjecture.

Empirical Law: *For Int the optimal truncation threshold τ is contained in the interval* $[0.125, 0.4]$.

Proof: Part of the result follows from the approximate formulas. For $\tau = 2^{-k}$ we obtain using the critical population size

$$FE = \frac{n}{k} * N^*(\tau_1) * 2^{\frac{k-1}{2}} \propto \frac{1}{\sqrt{\tau}\log(1/\tau)}, \quad k \geq 1 \tag{44}$$

The minimum lies between 0.125 and 0.4. ■

The empirical law has been investigated in detail by numerical experiments. The determination of the optimal population size by simulations is very difficult and error prone. We have done extensive simulations for two distributions generated by $OneMax$ and Int. The optimal population size is determined from the condition that from 1000 runs 900 find the optimum. The best numerical fit was obtained by using $\tau^{0.7}$ instead of $\tau^{0.5}$ for Int. For $OneMax$, $\tau^{0.8}$ gave a good fit.

These intensive simulations have been made to eliminate the truncation threshold as a free parameter. We formulate this important result as a rule.

Rule of Thumb:*The optimal truncation threshold for FDA is contained in* $0.125 \leq \tau \leq 0.4$. $\tau = 0.3$ *is a good choice.*

This result has been obtained by a mixture of theoretical results and numerical experiments. The same problem has been investigated for animal breeding. A discussion can be found in [15]. The empirical found result is the same. In most selection programmes that are at all efficient, I_τ lies between 1 and 2. This corresponds to $\tau = 0.4$ and $\tau = 0.06$.

8 The Factorization Problem

The FDA theory assumes an that an exact factorization of the probability distribution is given. Then convergence of FDA can be shown if the size of the population is large enough. But an exact factorization is not necessary for finding the optimum x_{opt}. If an approximate factorization $\tilde{p}(x,t)$ used for generating search points fulfils $\tilde{p}(x_{opt}) \geq p(x_{opt}$, then FDA will also converge to the optimum. This assertion is difficult to prove for a given distribution. Nevertheless it explains why FDA will also converge to the optimum for many approximate factorizations.

Another problem is that we assume that an ADF is explicitly given. For many physical problems this is the case. But there are of course problems where the structure of the ADF is unknown. In this case FDA has to estimate the probability model and its factorization. This techniques is called *learning of the probability model* in the theory of graphical models. It is an area of active research. FDA can easily be combined with a learning model. The requirements are that learning is not too expansive and that the learned probability model has a structure, which is very different from the ADF structure.

9 Conclusions

FDA combines evolutionary algorithms and simulated annealing. The theory is valid for discrete and continuous variables. By putting the emphasis on the estimation of distributions FDA also reconciles genetic algorithms with evolution strategies and evolutionary programming. The mathematical proof of convergence assumes that FDA is used with an exact factorization. But for many applications approximate factorizations are sufficient.

FDA can be combined with methods which “learn” the factorization. Therefore FDA can profit from any progress in this area.

Bibliography

[1] Aarts, E.H. & Korst, H.M. & van Laarhoven, P.J. (1997). Simulated Annealing. In Aarts, E. & Lenstra, J.K. (Eds.),*Local Search in Combinatorial Optimization.* Chichester:Wiley pp =121-136.

[2] Bäck, Th. & Schwefel, H.-P. (1993). An Overview of Evolutionary Algorithms for Parameteroptimization. *Evolutionary Computation*, 1:pp. 1-24.

[3] Baluja, S. & Davies, S. (1997). Using Optimal Dependency-Trees for Combinatorial Optimization: Learning the Structure of the Search Space. *Carnegie Mellon Report CMU-CS-97-107.*

[4] De Bonet, J.S.& Isbell, Ch. L. & Viola, P. (1997). MIMIC: Finding Optima by Estimating Probability Densities. In Mozer,M. & Jordan, M. & Petsche, Th. (Eds) *Advances in Neural Information Processing Systems 9.*

[5] de la Maza, M. & Tidor, B. (1993). An analysis of Selection Procedures with Particular Attention Paid to Proportional and Boltzmann Selection. In S. Forrest (Ed) *Proc. of the Fifth Int. Conf. on Genetic Algorithms* pp:124-131, San Mateo, CA: Morgan Kaufman.

[6] Frey, B.J. (1998). *Graphical Models for Machine Learning and Digital Communication.* Cambridge: MIT Press.

[7] Kargupta, H. & Goldberg, D.E. (1997). SEARCH, Blackbox Optimization, And Sample Complexity. In R.K. Belew & M. Vose (Eds.) *Foundations of Genetic Algorithms 4.* San Mateo, CA: Morgan Kaufman.

[8] Kargupta, H. (1997). *Revisiting The GEMGA: Scalable Evolutionary Optimization Through Linkage Learning.* Personal Communication.

[9] Kempthorne, O. & Folks, L. (1971). *Probability, Statistics and Data Analysis*. Ames, Iowa: The Iowa State University Press.

[10] Lauritzen, S.L. (1996) *Graphical Models.* Oxford:Clarendon Press.

[11] Mühlenbein, H. (1998). The Equation for Response to Selection and its Use for Prediction. *Evolutionary Computation,* 5:pp. 303-346.

[12] Mühlenbein, H. & Schlierkamp-Voosen, D. (1993). The science of breeding and its application to the breeder genetic algorithm. *Evolutionary Computation*, 1:pp. 335–360.

[13] Mühlenbein, H. & Schlierkamp-Voosen, D. (1994). The Theory of Breeding and the Breeder Genetic Algorithm. In J. Stender & E. Hillebrand & J. Kingdon (eds.), pp. 27-64, Amsterdam: IOS Press.

[14] Mühlenbein, H. & Mahnig, Th., Rodriguez Ochoa, A. (1999a). Schemata, Distributions and Graphical Models in Evolutionary Optimization. *to appear in Journal of Heuristics*

[15] Mühlenbein, H. & Mahnig, Th.(1999b). FDA - A scalable evolutionary algorithm for the optimization of additively decomposed discrete functions. *submitted for publication*

[16] Mühlenbein, H. & Paaß, G. (1996). From Recombination of Genes to the Estimation of Distributions I. Binary Parameters. In Voigt, H.-M et al. (eds.)*Lecture Notes in Computer Science 1141: Parallel Problem Solving from Nature - PPSN IV*, pp. 178-187, Berlin:Springer.

[17] Pelikan, M. & Mühlenbein, H. (1999). The Bivariate Marginal Distribution Algorithm, In Roy, R. & Furuhashi, T. & Chawdhry, P. K. (eds.), *Advances in Soft Computing - Engineering Design and Manufacturing*, pp. 521-535, Berlin:Springer-Verlag.