Drift, Diffusion and Boltzmann Distribution in Simple Genetic Algorithm

Hillol Kargupta
Department of Computer Science & Illinois Genetic Algorithm Lab.
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

Abstract

This paper presents a general diffusion model of a simple genetic algorithm. Unlike the similar previous efforts made for modeling mutation based genetic search, this work includes the effect of crossover by considering the dynamics of spatially averaged observables, under the influence of deterministic and stochastic effects. The analysis shows a possibility of generating Boltzmann distribution in the population. A noise reduction mechanism by means of controlling the distributional bias of the population is conjectured based on observations made on the entropic structure of the representational space.

1 Introduction

The evolution of the real world is characterized by complex self-organization at both macro and microscopic level. Present day science is far from any conclusive view about the underlying search process of natural self-organization. On the other hand, the artificial self-organizing systems often uses search algorithms, motivated by two streams of science: statistical mechanics and evolutionary genetics. Despite many differences between these two processes, fundamentally they seem to share the same mechanisms: a combination of deterministic and stochastic operators. The main accomplishment of the processes, based on the concepts of statistical mechanics of particles is associated with the generation of Boltzmann distribution. On the other hand, evolutionary computations are characterized by their fast pattern processing capability, using biases over the embedded representational space. Despite several attempts made for combining the virtues of their operators [22] [12] [18], very little work has been done to actually control the fundamental processes for generating the desired characteristics of the algorithm. Several authors have conjectured on the issue of asymptotic generation of Boltzmann distribution. However the efforts on generating canonical distribution asymptotically are restricted to cases with only mutation as the search operator [5] [3]. The proposition of controlling the mutation probability for reducing the noise and keeping the crossover completely out of the arena, does not really exploit the complete power of evolutionary computation.

Genetic algorithms belong to the category of evolutionary computation. They work from a randomly generated population of strings and makes use of the fitness information measure and a set of genetic operators to search for the goal. In this paper a framework for developing a continuous model of the physical processes governing the dynamics of a *simple genetic algorithm* is presented. After a general discussion on the issue of *optimal allocation of trial* in the context of GA, a continuous time, continuous space model for GA is developed in section 3. By transforming the observables of the system to some mean field, defined over a certain region in the space, continuous approximation is kept intact. This brings up the possibility of generating Boltzmann distribution over the current observable properties of the system. Section 6 proposes a different way of controlling the noise by adding *distributional bias* to the system. This becomes further clear and natural by looking at the entropy hierarchy in the sequence space. The propositions made in this paper need to be rigorously tested and the content should be considered as a preliminary report of the ongoing research.

2 Reliance on observation and allocation of trial

All natural self-organizing systems demonstrate a good deal of expertise in the process of organizing itself towards the optimal order or pattern, starting from a random state. The key issues in this dynamical process are,

- allocation of trial to the growing patterns for feature enhancement or inhibition.

0-8186-3420-0/93 $03.00

- feature transmission among the members of the system.

The problem of allocation of trial to a pattern optimally, has been addressed in the literature for long time. Most of the works done so far, in both symbolic and sub-symbolic paradigms, tried to optimize the allocation problem depending solely on the environmental pay-off received by a pattern. Even though it has been known that *pushing* hard towards reduced error state(i.e. with comparatively higher pay-off state) may cause serious problems, like convergence to local optima, unstability and so on, there has been very little theoretical formulation of the interaction of the strategy for allocation of trial with the process of pattern formation.

One of the most widely cited result as described in [14] [11], says that, if the expected pay-off of some two patterns are μ_1 and μ_2 with variances σ_1 and σ_2 respectively, and we do not know beforehand which one is associated with the higher expected pay-off (say μ_1), for $\mu_1 \geq \mu_2$, then the optimal number of trial given to the observed better pattern is,

$$n_1^* \cong \sqrt{8\pi b^4 \ln N^2}\exp^{\frac{n_2^*}{2b^2}} \qquad n_2^* = N - n_1^* \quad (1)$$

where, N=total number of trials and b=$\sigma_1/(\mu_1+\mu_2)$. Basically, this means that the observed best should be given little more than an exponentially increasing number of trials. This finding has been used as a kind of unrealizable bound in practice by genetic algorithms(GA) [14][11]. Clearly, this exponential reliance on observation is a function of the signal to noise ratio(i.e. the ratio of the expectation and variance terms). Despite the presence of *signal to noise* ratio terms in this non-equilibrium strategy for optimal allocation of trial, canonical GA does not have any explicit mechanism to control its reliance over the observed fitness. In fact in canonical GA, observed best pattern experiences exponentially increasing number of trials in the initial stage and gradually the rate reduces as the population average fitness increases. GA is a population based approach and thereby has more information about the state space in the initial state than that of any pointwise search algorithm. Even then, it is an open question that how justified is this exponential reliance through out its non-equilibrium trajectory. A recent effort [13] has shown that if the initial population size is large enough to make sure large signal to noise ratio, the exponential reliance on initial observation can be guaranteed to be near optimal. However, this analysis does not say anything about the reliability of this strategy under the presence of operator induced noise during the long course of non-equilibrium trajectory.

For a system in equilibrium, the Boltzmann distribution is given by,

$$n_x = \frac{N\exp^{\beta f(x)}}{\sum_y \exp^{\beta f(y)}} \quad (2)$$

where n_x is the number of instances of the state x in the *Boltzmann population* of total size N, with environmental pay-off f(x) and β is an internal property of the system. Equation 2 is presented in a general way in order to facilitate proper interpretation in the current topic. This can be interpreted in the present context, as the number of trial given to a particular configuration, depends exponentially on its environmental pay-off, and also on a parameter β, which is essentially a measure of the noise prevailing in the system. β plays a very important role in all natural self-organizing systems. For example, in thermodynamics, $\beta = \frac{1}{kT}$ where k is the *Boltzmann constant*, T is the absolute temperature and it is well known how temperature controls the thermodynamics of the systems. The nice statistical properties of Boltzmann distribution has not gone unnoticed. One of its most famous realization in the field of search and information processing, is a general purpose search algorithm, Simulated Annealing (SA) [16], [1]. SA asymptotically generates the Boltzmann distribution by repeated application of a perturbation operator to generate neighbors of the current state and using the so called *logistic probability* to accept or reject that transformation. It follows a cooling schedule for governing its reliance on observation by explicitly controlling the logistic probability. The simulated cooling is basically the noise reduction mechanism.

The issue of noise reduction is very important in search. Noise can be carefully used to explore unknown regions of the search space [17]. It will be of little use reexploring the already visited region of the space. In other words, noise should be cleverly produced by the search operators for exploring the unknown regions not the known ones. This is the main essence of the present work. In the following sections, an effort will be made towards incorporating this philosophy in traditional Genetic Algorithm.

3 Drift, Diffusion and Turbulence

From the point of view of statistical mechanics drift and diffusion are two aspects of the behavior of an ensemble of particles dominated by random thermal motion. Even though they are not fundamentally different processes, drift is usually referred to the motion

of particles under the influence of some external force and diffusion to the motion resulting from the spatial concentration gradient of the particles. One interesting feature of drift-diffusion systems is that at equilibrium state they generate Boltzmann distribution. Immediate example of this can be found in earth atmosphere, semiconductor materials and so on. Drift and diffusion are also present in both natural and artificial information processing systems. However in this paradigm, sometimes they are present in an abstract sense. Drift may be identified as the feature enhancement or inhibition mechanism as an immediate result of the environmental pay-off. Diffusion may come into the picture through the random or stochastic nature of these systems. In terms of randomness, diffusion can be defined as an irreversible process by which members of an ensemble *move* out within a given space(which could be either an ordinary Euclidean or some abstract space) according to the individual random motion. A tighter link can be drawn by comparing a simple one dimensional random walk and a physical diffusion process. After a large number of steps, a one dimensional isotropic random walk gives rise to a Gaussian probability distribution.

The transportation of particles in a diffusive system is known to be governed by Fick's law of diffusion. This basically says that the particle flux in a certain direction is proportional to the particle concentration gradient along that direction. Fick's second law of diffusion is,

$$\frac{\partial}{\partial t}C(x,t) = \frac{\partial}{\partial x}\left(D\,\frac{\partial}{\partial x}C(x,t)\right) \qquad (3)$$

where D is the diffusivity(assumed to be constant); x and t are the spatial and time coordinates respectively. The solution to the Fick's law gives the particle distribution,

$$C(x,t) = \frac{n}{2\sqrt{\pi Dt}}\exp\left(-x^2/4Dt\right)$$

where, n is the number of particles in a unit area, concentrated at x = 0 This is basically the same Gaussian distribution as produced by random walk process. This similarity has been exploited in the field of modeling repeatedly. Population geneticists also picked up this in order to model the behavior of a population structure along the spatio-temporal landscape. It is often very difficult to come up with a closed form solution of the complex models of evolutionary process. Even for models with an explicit solution, very little insight can be gained by looking into its complicated expressions. That is the reason why diffusion models have enjoyed so much of attention, ever since Fisher [8] and Wright [24] introduced them. Diffusion models have been found to provide good approximations for some restricted population genetics problems [7][23]. The fundamental bottleneck of this approach lies in the continuous time, continuous space and markovian approximation of the basically discrete process. Since nearly all the work done in theoretical population genetics is in the gene frequency space, the very survival of the diffusion approximation depends on the restricted application to cases with small changes in gene frequencies per unit of time.

Another interesting kind of diffusion shows up in *turbulence*. Turbulence can be viewed as a collection of *eddies* of different kinds. An eddy is basically a set of particles, sharing some common flow features, such as *characteristic velocity*, *characteristic size* and so on. Eddies can be of different sizes and mutually overlapping. The particles belonging to a particular eddy have the common average velocity along with their own fluctuating components. One of the major differences between molecular agitation and turbulent movement is the difference in scale; the order of a single turbulent movement is usually much larger than that of the molecular mean free path. This introduces a fundamental problem with simple use of diffusion equation for modeling the random turbulent phenomena. Since the typical time and length scales of turbulence are of the same order of magnitude as typical time and length scales of observation, naive incorporation of the random walk model treads on thin ice. This fundamental problem can be avoided if we look at turbulence in a different way. In stead of observing a single particle, if we look at the characteristics averaged over spatial scale, by increasing the sample size, the diffusion approximation survives. In order to achieve this in practice, some times a very large sample size is needed. This can be further addressed by introducing a temporal average of the spatially averaged features of a reasonable amount of sample. Basically, the idea is to define some smooth observables of the process, by applying spatio-temporal averaging. These techniques brings in a very interesting way to deal with systems, having large fluctuations. In the next section, a close parallel to turbulent diffusion will be drawn in the GA landscape and we will further observe that how a traditional metric in GA information processing fits very well into these ideas.

4 Brief review of simple genetic algorithm

Genetic Algorithm(GA) is a general purpose search algorithm, motivated by the Darwinian theory of nat-

ural selection and genetics. GA is a population based approach in contrast to the conventional point wise search techniques. It uses operators like selection, crossover and mutation, which are quite similar to their natural counterpart, and the observed environmental pay-off information, in order to search for optimal structure from a randomly generated population of structures.

The basic mechanism of GA can be formally expressed as follows:

1. **Initialization:** At time $t = 0$, randomly generate N strings, $\mathbf{s_1}, \mathbf{s_2}, \cdots \mathbf{s_N}$. Each of these strings is a sequence of random variables, $y_1, y_2, \cdots y_l$, which could be either discrete or continuous. For the rest of this paper, we will assume them to be discrete. There exists a measure of fitness of $\mathbf{s_i}$, $\pi(\mathbf{s_i})$.

2. **Selection + Crossover:** Select two strings from the population at time t, in such a way that the selection probability of a string $\mathbf{s_i}$ is,

$$p(\mathbf{s}_i, t) = \frac{\pi(\mathbf{s}_i)}{\sum_{j=0}^{N} \pi(\mathbf{s}_j)} \tag{4}$$

Apply crossover operator on them, with a certain probability p_c. There are several types of crossover operators prevailing in the GA literature. The discourse in sequel, addresses only a generic kind of uniform crossover, which will be relevant to our future discussion. Crossover picks up some k entries out of the l genes from one parent and rest of the $l-k$ genes from the other parent. Keep the two newly generated strings in a temporary population set. Repeat this process until the population size becomes N.

Consider the string $\mathbf{s}_i = y_1, y_2, \cdots y_l$. The string $\mathbf{s}_j(y_1, y_2, \cdots y_k, y''_{k+1}, y''_{k+2}, \cdots y''_l)$ has $y_1, y_2, \cdots y_k$ entries same as $\mathbf{s_i}$ and another string $\mathbf{s}_k(y'_1, y'_2, \cdots y'_k, y_{k+1}, y_{k+2}, \cdots y_l)$ has $y_{k+1}, y_{k+2}, \cdots y_l$ entries same as $\mathbf{s_i}$. Denote the set of all possible strings having the $y_1, y_2, \cdots y_k$ entries common by $\mathbf{s}(y_1, y_2, \cdots y_k)$ and similary define $\mathbf{s}(y_{k+1}, y_{k+2}, \cdots y_l)$. The probability of generating the string $\mathbf{s_i}$, by crossing over between a parent from, taken from $\mathbf{s}(y_1, y_2, \cdots y_k)$ and the other from $\mathbf{s}(y_{k+1}, y_{k+2}, \cdots y_l)$ is,

$$\begin{aligned} P_g(\mathbf{s_i}) &= p^k(1-p)^{l-k}\, \overline{p}(\mathbf{s}(y_1, y_2, \cdots y_k)) \cdot \\ &\quad \overline{p}(\mathbf{s}(y_{k+1}, y_{k+2}, \cdots y_l)) \end{aligned} \tag{5}$$

where,

$$\begin{aligned} \overline{p}(\mathbf{s}(y_1, y_2, \cdots y_k)) &= \sum_{y_{k+1}, y_{k+2}, \cdots y_l} f_s(\mathbf{s}) \cdot \\ &\quad \pi(\mathbf{s}(y_1, y_2, \cdots y_k)) \end{aligned}$$

$f_s(\mathbf{s})$ stands for the probability mass function. Now summing this over every possible combination of $y_1, y_2, \cdots y_k$ out of the set $\{y_1, y_2, \cdots y_l\}$ and again summing over all possible values of $k \varepsilon \{0, 1, 2, \cdots, l\}$, we get the probability of generating the string $\mathbf{s_i}$ from the population at time t [20],

$$\begin{aligned} P_g(\mathbf{s_i}) &= \sum_{k=0}^{l} p^k(1-p)^{l-k} \cdot \\ &\quad \sum_{y_1, \cdots y_k} \overline{p}(\mathbf{s}(y_1, \cdots y_k)) \overline{p}(\mathbf{s}(y_{k+1}, \cdots y_l) \end{aligned} \tag{6}$$

One important aspect of the behavior of crossover is the choice of k entries out of $y_1, y_2, \cdots y_l$. Clearly, anisotropy can be introduced into the search by using a crossover, which has biases towards certain groups of entries. *Distributional bias* is one of the biases which can be introduced by crossover. This simply means that crossover has a bias towards a certain value of k. *Positional bias* is another example of crossover biases. More detailed discussion about crossover biases can be found in [6].

3. **Mutation:** Mutation is usually profile operator and it changes the entries $y_1, y_2, \cdots y_l$ with a very low probability. Mutation is applied on the population of new strings generated by crossover.

4. Replace the old population by the new strings obtained after applying mutation. Set $t = t + 1$ and go back to step number 2, unless the stopping criteria is satisfied.

Before we start any formal analysis of dynamics of GA, it is quite natural to ask a simple question - what are the observable features of this evolutionary process? The first shot answer could be the string growth. Simple observation of the status(i.e. the number of copies) of every possible string in the population introduces a philosophical question. All natural self-organizing systems are characterized by their gradual increase in complexity. The process of *crystallization* is characterized by distinct stages - *nucleation* and *grain growth* . Turbulent flow experiences gradual evolution of large eddies by the merger of smaller eddies.

The question is if we want to observe the evolutionary process of such a self-organizing phenomena, is it not good enough to look at the growing patterns(i.e. the notion of structure introduced so far in the system), by means of some complexity metric? The answer is, yes; but it is difficult of define a good complexity metric for complex systems, since the notion of complexity is yet to be well defined. GA uses a simple sequence representation of the information. A good measure of complexity in sequence space can be defined to be one which recognizes the evolution of *markov dependencies* among the symbols in a sequence. Shanon's entropy could be an example of that. John Holland introduced the notion of a kind of *schema*[14] for analyzing the information processing in GA, which shares the same philosophical view. A schema is a similarity template describing a subset of strings with similarities at certain string positions. For example if $1, 0$ be an alphabet and $*$ is a don't care character, which can match with any letter in $1, 0$ then the schema $1{**}1{**}$ represents the set of strings having letter 1 at the first and fourth position; string 100101 is an instance of that schema, but 010101 is not. The *order* of a schema is the number of fixed letters, i.e. symbols other than $*$. In following discourse we will see how the idea of schema can be used to define observables, which capture the idea of structure among sequences.

5 A drift-diffusion model for GA

The main problem with the discrete analysis of GA is essentially the same as that experienced by population geneticists about a century back. This is simply very difficult to deal with. Several previous attempts made along this line, resulted in some discrete time models of GA[25][20]. Unfortunately, it is very difficult to boost our understanding about the process of pattern formation and the asymptotic behavior of the process from these models. In the following discourse an effort will be made to model GA as a continuous time, continuous space, markovian process. If the population is large enough, spatial continuity could be a reasonable approximation. But the continuous time approximation needs careful attention. Crossover operator usually destroys the participating strings with a very high probability. These large fluctuations can be scaled down by applying the spatial smoothing technique, as we have earlier noticed in the case of turbulent diffusion.

For a moment consider a hypothetical continuous time, continuous space GA. Define a n-dimensional space, X, where a point $\mathbf{x} = (x_1, x_2, \cdots x_n)$; x_i stands for the proportion of *i-th* string present in the population at a certain time. If the length of each string is l, then $n = 2^l$ in case of binary alphabets. Let $\phi(\mathbf{x_0}, \mathbf{x}; t)$ be the conditional probability density that the population will be at point $\mathbf{x}$ at time t, given the initial state as $\mathbf{x_0}$ at time $t = 0$. If $\psi(\delta\mathbf{x}, \mathbf{x}; \delta t, t)$ be the the transition probability from $\mathbf{x}$ to $\mathbf{x} + \delta\mathbf{x}$ during the time interval $(t, t + \delta t)$,

$$\phi(\mathbf{x_0}, \mathbf{x}; t) = \int \phi(\mathbf{x_0}, \mathbf{x} - \delta\mathbf{x}; t)\psi(\delta\mathbf{x}, \mathbf{x} - \delta\mathbf{x}; \delta t, t)d(\delta\mathbf{x}) \tag{7}$$

The integrand can be expanded as,

$$\phi(\mathbf{x_0}, \mathbf{x} - \delta\mathbf{x}; t + \delta t)\psi(\delta\mathbf{x}, \mathbf{x} - \delta\mathbf{x}; \delta t, t) = \phi\psi - \sum_i \delta x_i \frac{\partial(\phi\psi)}{\partial x_i} + \sum_{i,j} \frac{\delta x_i \delta x_j}{2!} \frac{\partial^2(\phi\psi)}{\partial x_i \partial x_j} - \cdots \tag{8}$$

where ϕ and ψ are the same as $\phi(\mathbf{x_0}, \mathbf{x}; t)$ and $\psi(\delta\mathbf{x}, \mathbf{x}; \delta t, t)$ respectively. Substituting equation 7 into equation 8, we can come up with the general *diffusion equation*,

$$\frac{\partial\phi}{\partial t} = -\sum_i \frac{\partial}{\partial x_i}(\Theta_i(x_i, t)\phi) + \frac{1}{2}\sum_{i,j} \frac{\partial^2}{\partial x_i \partial x_j}(\Gamma_{ij}(x_i, x_j, t)\phi) \tag{9}$$

Θ_i and Γ_{ij} are defined as,

$$\lim_{\delta t \to 0} \frac{1}{\delta t} \int (\delta x_i)\psi d(\delta x_i) = \Theta_i(x_i, t)$$

$$\lim_{\delta t \to 0} \frac{1}{\delta t} \int \delta x_i \delta x_j \psi d(\delta x_i) = \Gamma_{ij}(x_i, x_j, t)$$

$\Theta_i(x_i, t)$ and $\Gamma_{ij}(x_i, x_j, t)$ can be replaced by the mean (Θ_i) and covariance(Γ_{ij}) terms of the increment in x_i, per generation. They are usually known as the *drift* and *diffusion* coefficients, respectively.

In the case of real GA, things are not all that simple. The above diffusion equation reduces to instantaneous diffusion. The instantaneous probability flux along i-th direction is,

$$F_i = \Theta_i\phi - \frac{1}{2}\sum_j \frac{\partial}{\partial x_j}(\Gamma_{ij}\phi) \tag{10}$$

Equation 9 now becomes,

$$\frac{\partial\phi}{\partial t} = -\nabla \cdot \mathbf{F} \tag{11}$$

So far in this section, we have considered every string as a separate entity, without considering the embedded representational space. At this point we would like to define the system observables more precisely. As earlier hinted, the key concept is how we look at the process. If we carefully look at the random fluctuations caused by crossover in the embedded

representational space, we will notice that the magnitude of the fluctuation, defined over some similarity metric(e.g. hamming distance) depends on the number of entries exchanged by crossover (the variable k in equation 6). An example will further clarify the issue.

If the crossover exchanges 2 entries between the parents and since the 2 entries are chosen from each parent randomly, there are $\binom{l}{2}$ ways of choosing the entries. So if the two parents are completely dissimilar, every crossover may introduce fluctuation over a subset of $2\binom{l}{2}$ strings; remember that each of these strings corresponds to a specific coordinate in our frame of reference. Similarly, exchanging 4 entries may introduce fluctuation over even a larger subset, defined over $2\binom{l}{4}$ coordinates. Repeated number of application of crossover among the population members results in considerable amount of fluctuations over the expected change along different coordinates, defining such subsets. Each of these subsets can be identified as the *schema subset*, defined in section 4. Depending on the value of k such subsets of several sizes may be defined. Since the fluctuations along a particular coordinate is very high, we can simply observe the average properties over these subsets and see what kind of diffusion equation governs them. Note that only spatial averaging is used here, in contrast to the spatio-temporal averaging of turbulence. The expected change in proportion of a string can be written as,

$$\Theta_i = \overline{\Theta} + \Theta_i'$$

where $\overline{\Theta}$ is the average of the expected increment over a schema subset and Θ_i' is the random fluctuation component of the increment in x_i. Substituting equation in and averaging over the schema subset,

$$\overline{F_i} = \overline{\Theta_i}\ \overline{\phi} + \overline{\Theta_i'\phi} - \frac{1}{2}\sum_j \frac{\partial}{\partial x_j}(\Gamma_{ij}\overline{\phi}) \qquad (12)$$

The first term of the right hand side of the above equation is the *advection* term. The third term corresponds to the diffusion because of the low level noise added by selection and mutation; this is quite similar to the molecular diffusion contribution in case of turbulent flow. The middle term is solely due to the turbulence like effects, caused by the crossover operation. $\overline{\phi}$ gives the probability density that the population will be within the observed volume, defined by the above explained subsets. In order to make the middle expression more meaningful, we need to take one more conceptual leap. Consider the case of pure turbulent diffusion, i.e. when the first and third terms of equation 12 are absent.

Let $\eta(\delta\mathbf{x}, t)$ be the p.d.f. of the change in $\mathbf{x}$ because of the pure turbulent diffusive action. As we have seen earlier, the generation of strings is binomial in nature. This leads us to assume $\eta(\delta\mathbf{x}, t)$ to be normal. We assume under pure diffusive action the distribution of δx_i -s are jointly as well as separately normal. Now it can be easily shown that,

$$\frac{\partial\overline{\phi}}{\partial t} = \sum_{i,j}\frac{1}{2}\Xi_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(\overline{\phi}) \qquad (13)$$

where Ξ_{ij} is the covariance term introduced by the crossover. Averaging equation 11 and using equations 12 & 13,

$$\begin{aligned}\frac{\partial\overline{\phi}}{\partial t} &= -\sum_i \frac{\partial}{\partial x_i}\overline{\Theta_i}\ \overline{\phi} + \sum_{i,j}\frac{1}{2}\frac{\partial^2}{\partial x_i\partial x_j}(\Xi_{ij}\overline{\phi}) \\ &\quad + \frac{1}{2}\sum_{i,j}\frac{\partial^2}{\partial x_i\partial x_j}(\Gamma_{ij}\overline{\phi}) \qquad (14)\end{aligned}$$

In the equilibrium state, $\frac{\partial\overline{\phi}}{\partial t} = 0$.

$$-\sum_i \frac{\partial}{\partial x_i}\left(\overline{\Theta_i\phi} - \frac{1}{2}\sum_j\frac{\partial}{\partial x_j}(\Gamma_{ij}+\Xi_{ij})\overline{\phi}\right) = 0$$

The combined covariance matrix, $(\Gamma_{ij}+\Xi_{ij})$ asymptotically approaches to some constant value. The expression within the parenthesis is basically the *flow of probability* along the $i-th$ direction. If there exists a stable equilibrium distribution, such that the probability flux is zero at every point, i.e.,

$$\left(\overline{\Theta_i\phi} - \frac{1}{2}\sum_j(\Gamma_{ij}+\Xi_{ij})\frac{\partial}{\partial x_j}\overline{\phi}\right) = 0 \qquad (15)$$

After rearranging and using the vector notation,

$$\left[\frac{\partial}{\partial\mathbf{x}}(\ln\overline{\phi}(\mathbf{x}))\right] = [\Gamma+\Xi]^{-1}2\overline{\Theta} \qquad (16)$$

where $[\Gamma+\Xi]^{-1}$ is the inverse of the combined covariance matrix, $[\Gamma+\Xi]$. Note that $[\Gamma+\Xi]$ is not supposed to be singular.

$$\frac{\partial}{\partial\mathbf{x}}(\ln\overline{\phi}(\mathbf{x})) = \begin{bmatrix}\frac{\partial}{\partial x_1}(\ln\overline{\phi}(\mathbf{x}))\\ \frac{\partial}{\partial x_2}(\ln\overline{\phi}(\mathbf{x}))\\ \vdots\end{bmatrix}\ ;\ \overline{\Theta} = \begin{bmatrix}\overline{\Theta_1}\\ \overline{\Theta_2}\\ \vdots\end{bmatrix}$$

Solving equation ,

$$\overline{\phi}(\mathbf{x}) = \exp^{[\Gamma+\Xi]^{-1}\int 2\overline{\Theta}d(\mathbf{x})} \tag{17}$$

Writing $\int 2\overline{\Theta}d(\mathbf{x})$ as a potential function $\xi(\mathbf{x})$,

$$\overline{\phi}(\mathbf{x}) = \exp^{[\Gamma+\Xi]^{-1}\xi(\mathbf{x})} \tag{18}$$

This provides a theoretical basis for generating a Boltzmann distribution, in the mean field generated by the schema subsets.

6 Entropy reduction by controlling noise

The process of self-organization can be viewed as a noise driven, entropy reduction process. Entropy reduction itself may be not be very informative, unless we specify how it is reduced. The covariance matrices in equation 18 are the representatives of the noise, prevailing in the system. Entropy of a system provides another kind measure about the noise level. For the sequence of discrete random variables, $\mathbf{s} = y_1, y_2, \cdots y_l$, l-tuple Shanon's entropy can be defined as,

$$H_l = -\sum_{i_1}\sum_{i_2}\cdots\sum_{i_l} p(y_{i_1}y_{i_2}\cdots y_{i_l})\log p(y_{i_1}y_{i_2}\cdots y_{i_l})$$

If each of the entries of the l-tuples depends on m other entries, i.e. if it is an mth order Markov source,

$$\begin{aligned} H_l &= H_1 + H_M^1 + \cdots + H_M^{m-1} + (l-m)H_M^m \\ &= H_M^0 + H_M^1 + \cdots + H_M^{m-1} + (l-m)H_M^m \end{aligned}$$

where

$$\begin{aligned} H_M^m &= -\sum_{i_1}\sum_{i_2}\cdots\sum_{i_{(m+1)}} p_{i_1}p_{i_1i_2}\cdots p_{i_1i_2\cdots i_{(m+1)}} \cdot \\ &\quad \log p_{i_1i_2\cdots i_{(m+1)}} \end{aligned}$$

The divergence from the maximum entropy state, can be expressed as a summation of the change in entropy, because of the divergences of every i-tuplets(i=1$\cdots$m) from equiprobability. So the divergence from the maximum entropy state, i.e. the information stored in the space[10], is

$$D^m = \sum_{i=0}^{m} d_i \tag{19}$$

where

$$\begin{aligned} d_i &= H_{max} - H_M^0 \qquad \text{if } i = 0 \\ &= H_M^{i-1} - H_M^i \qquad \text{otherwise} \end{aligned}$$

where, H_{max} is maximum entropy state corresponding to equiprobable distribution. Basically, d_i-s are the divergences of the distribution of an i-th order Markov source from equi-probability. Equation 19 clearly shows the hierarchical nature of entropy in the sequence space. As patterns or order starts growing in a system, it gradually moves down the entropy scale, which is divided into several levels, corresponding to the order of the pattern, i.e. the order of the Markov source.

It is very interesting to note that, natural evolutionary self-organization does critically control the distribution of different order of patterns, i.e. the d_i-s. Several researchers have pointed out this fact elsewhere [10][4]. It has been found that, highly organized organisms, have a relatively much smaller value of d_0 than that of primordial organisms. This strategy is usually described to be the key factor behind maintaining the genetic diversity. This indicates that, in order to control the growth of patterns at every stage, the state transition operators should have some mechanisms to control every order of d_i-s. As we have seen earlier, crossover operator can introduce two kinds of biases in the search process - 1) *positional bias* 2) *distributional bias*. It can be shown that the noise prevailing in the system can be brought in the level corresponding to d_i by adding i-th order *distributional bias* to the crossover operator. It is even more interesting that crossover does not change the *marginal probability distribution* and thereby keeps the d_0 value to its minimum.

Let us go back to our analysis of crossover in section 1. It should be noted that exchanging k entries is equivalent to exchanging $l-k$ entries. It can be shown from equation 6 that when k randomly varies between 0 and l repeated application of crossover decorrelates the variables, $y_1, y_2, \cdots y_l$[9][25][2][20]. Now removing the summation over k from equation 6 we get,

$$\begin{aligned} P_g(\mathbf{s_i}) &= p^k(1-p)^{l-k} \sum_{y_1,y_2,\cdots y_k} \overline{p}(\mathbf{s}(y_1, y_2, \cdots y_k)) \cdot \\ &\quad \overline{p}(\mathbf{s}(y_{k+1}, y_{k+2}, \cdots y_l)) \end{aligned} \tag{20}$$

It can be shown that the above equation generates a k-th order markov dependencies in the population. This brings up the possibility of noise reduction by means of controlling the *distributional bias* of the crossover operator. Intuitively this draws a parallel between the temperature decreasing schedule of either of real or simulated annealing process, with the essential difference of noise reduction, while preserving the process of structure formation in the representational space.

However more rigorous analysis of noise reduction schedule and crossover behavior are required before completely defining the algorithm.

7 Conclusion

In this work, a modest step has been made for formulating GA as a diffusion processes. This in turn opens up a possibility of generating Boltzmann distribution in the population, by carefully designing the crossover operator. Moreover this work shows that the crossover operator of GA has an effect similar to eddy diffusion, which is usually much more effective than molecular diffusion. However, the proposed noise reduction schedule, by controlling the distributional biases of the crossover needs more analytic attention, before rigorously defining the algorithm.

8 Acknowledgement

The author would like to acknowledge the support and help provided by Prof. David E. Goldberg for this work.

References

[1] Aarts, E. and Korst, J. *Simulated Annealing and Boltzmann Machines: a Stochastic Approach to Combinatorial Optimization and Neural Computing*, Chichester: John Wiley & Sons, 1989.

[2] Booker, L. B. "Recombination distributions for genetic algorithms ", To appear in Whitely D. (Ed.) *Foundation of Genetic Algorithms*, Vale, 1992.

[3] Boseniuk, T., Ebeling V. & Engel A. " Boltzmann and Darwin strategies in complex optimization, *Physics Letters* A, 125, no. 6/7, 307-310, 1987.

[4] Brooks, D. R. & Wiley, E. O. *Evolution as entropy: toward a unified theory of the biology*, Chicago: University of Chicago Press, 1986.

[5] Ebeling W. "Diffusion and models of evolution process", *Journal of Statistical Physics*, 37, no. 3/4, 369-384, 1984.

[6] Eshelman, L. J., Caruana, R. A., & Schaffer J. D. "Biases in the crossover landscape", *Proceedings of the Third International Conference on Genetic Algorithms*, 10-19, 1989.

[7] Feller, W. Diffusion processes in genetics, *Proc. Second Berkeley Symp. Math. Stats. Prob.*, University of California Press, 227-246, 1951.

[8] Fisher, R. A. *The genetical theory of natural selection*. Oxford University Press, 1930.

[9] Geiringer, H. "On the probability theory of linkage in Mendelian heredity", *Annals of Mathematical Statistics*, 15, 25-57, 1944.

[10] Gatlin L. L. *Information Theory and the Living System*, Columbia University Press, 1972.

[11] Goldberg D. E. *Genetic Algorithms in Search, Optimization and Machine learning*, Addison-Wesley Publishing Company Inc., 1989.

[12] Goldberg D. E. (1990). "A Note on Boltzmann Tournament Selection for Genetic Algorithms and Population Oriented Simulated Annealing", *Complex Systems* , 4, 445-460, 1990.

[13] Goldberg D. E., Deb, K. & Clark, J. H. *Genetic algorithms, noise and the sizing of populations* (IlliGAL Report 91010). Urbana: University of Illinois, Illinois Genetic Algorithm Lab, 1991.

[14] Holland, J.*Adaptation in Natural and Artificial Systems*, Ann Arbor, The University of Michigan press, 1992.

[15] Kimura, M. *Diffusion models in population genetics*, Methuen's Riview series in Applied Probability, 1964.

[16] Kirpatrick, S., Gelatt, C. D. & Vecchi, M. P. "Optimization by simulated annealing". *Science*, 220(4598), 671-680, 1983.

[17] Landauer R. "Noise-activated escape from metastable states: an historical view", Moss, F. & McClintoch, P. V. E., (Eds.) *Noise in Nonlinear Dynamical Systems*, vol.1, Cambridge University Press, 1989.

[18] Mahfoud S. W. & Goldberg D. E. "Parallel Recombinative Simulated Annealing: A Genetic Algorithm", *Parallel Problem Solving from Nature* (to be published), 1992.

[19] Okubo, A. *Diffusion and ecological problems: mathematical models*, Springer-Verlag, 1980.

[20] Qi, X. & Palmieri, F. *general properties of genetic algorithms in the euclidean space with adaptive mutation and crossover*, Technical Report EE-92-04, Dept. of Electrical and Systems Engg., University of Connecticut, 1992.

[21] Pai, S. *Viscous flow theory II - turbulent flow*, D. Van Nostrand Company Inc., 1957.

[22] Sirag, D. J., & Weisser, P. T. (1987). "Towards a unified thermodynamic genetic operator". *Genetic algorithms and theor applications: proceedings of the Second International Conference on Genetic Algorithms*, 116-122, 1987.

[23] Svirezhev Y. M. & Passekov V. P. *Fundamentals of Mathematical Evolutionary Genetics*, Kluwer Academic Publishers, 1989.

[24] Wright, S. "The differential equation of the distribution of gene frequencies". *Proceedings of National Academy of Science*, 31, 382-389, 1945.

[25] Vose, M. D. & Liepins, G. E. "Punctuated equilibria in genetic search", *Complex systems*, 3, 31-44, 1991.