Finite Populations Induce Metastability in Evolutionary Search

Erik van Nimwegen,[†]
James P. Crutchfield,[†‡] and
Melanie Mitchell[†]
[†]Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501
[‡]Physics Department, University of California, Berkeley, CA 94720-7300

Physics Letters A, 229 (2), 144-150 (1997).

Abstract

We introduce an analytical model that predicts the dynamics of a simple evolutionary algorithm in terms of the flow in the space of fitness distributions. In the limit of infinite populations the dynamics is derived in closed form. We show how finite populations induce periods of stasis—"fitness epochs"—and rapid jumps—"innovations". The analysis identifies the epochs with the flow's metastable fixed points and gives exact predictions of epoch fitness level and population distribution.

Evolutionary search comprises a set of optimization and learning techniques—based loosely on ideas from biological evolution—in which individuals in a population (typically candidate solutions to a given problem) are selected for replication based on a "fitness" measure defined by some externally imposed criterion. Genetic variation is introduced into the population by stochastically modifying the selected individuals. In practice evolutionary search methods have often been applied successfully to optimization and learning problems in in poorly understood high-dimensional problem spaces. For overviews of such methods and their applications, see, e.g., [1]. In the following we analyze one particular evolutionary search method: a simplified genetic algorithm (GA).

One common feature of the dynamics of evolutionary search is metastable behavior—the average fitness in the population goes through periods of stasis, which we will call "fitness epochs", alternating with rapid jumps in average fitness, which we will call "innovations". Such behavior has been reported in many applications of evolutionary search as well as in models of evolution (e.g., [2]). As has often been found, the length, location, and shape of these fitness epochs and innovations depend on different control parameters of the evolutionary method, such as population size and mutation and crossover rates. This dependence is complicated and, as yet, poorly understood.

There has been a moderate amount of theoretical work on evolutionary algorithms. The approach of Vose, Nix, and Liepins, for example, describes the state of a genetic algorithm at a certain time by a vector on the simplex of a high-dimensional Euclidean space. Each dimension of this space represents either a certain genome [3] or the state of the population as a whole [4]. The GA dynamics is specified by a nonlinear matrix operator that acts on this vector to produce the state at the next time step. Although this formalism exactly captures the detailed "microscopic" GA dynamics, in practice the large size of these matrices makes it impossible to obtain quantitative results. Here, a matrix operator will be constructed that is similar in spirit to the Vose *et al.* operators but that has vastly reduced dimensionality since it acts on vectors representing fitness distributions. This makes a quantitative analysis of this operator possible, allowing specific quantitative predictions to be made about the GA's behavior.

Another theoretical approach is a formalism developed by Prügel-Bennett, Shapiro, and Rattray that uses a statistical mechanics approach to analyze the dynamics of genetic algorithms [5]. In their formalism the focus is also on the evolution of fitness distributions, but generally only the *average* evolution of the first few cumulants of the fitness distribution are studied. This averaging of the dynamics over a large number of runs makes it impossible to describe the epoch-innovation structure of the dynamics in which we are interested.

The GA's population will consist of bit strings of length $L = NK$. The fitness function f considers each string $\mathbf{s}$ to consist of N contiguous blocks of K bits. For each block of length K, f specifies a particular desired bit configuration (schema). Without loss of generality, this desired configuration can be taken to be K $1s$ for each block. The fitness $f(\mathbf{s})$ of a string $\mathbf{s}$ is then simply the number of blocks in string $\mathbf{s}$ that consist of K ones. Thus, $0 \leq f(\mathbf{s}) \leq N$.

This type of fitness function, a so-called "Royal Road" function, was initially designed to address questions about the processing and recombination of schemata by GAs. Such block-based fitness functions were thought to lay out a "royal road" on which genetic algorithms would optimally search. In fact, the GA dynamics is substantially more complicated than originally hypothesized for these idealized fitness functions [6]. Here we use f because it is simple enough to be analyzed and yet the dynamics of the GA on this fitness function exhibits the epochal dynamics so often encountered in evolutionary search.

Our analysis assumes the following simplified genetic algorithm. Initially, the GA generates a population $\mathcal{P}_0$ of M bit strings of length $L = NK$, chosen with uniform probability from the space of all L-bit strings. It then evaluates the fitness $f(\mathbf{s})$ of each string $\mathbf{s} \in \mathcal{P}_0$. Next, it creates a new population of M strings, by selecting strings from the current population with replacement, and with probability proportional to fitness. Finally, it mutates each selected string at each site with a fixed probability q. At this point a new population $\mathcal{P}_1$ has been created and the cycle starts over by re-evaluating the fitnesses of the new strings.

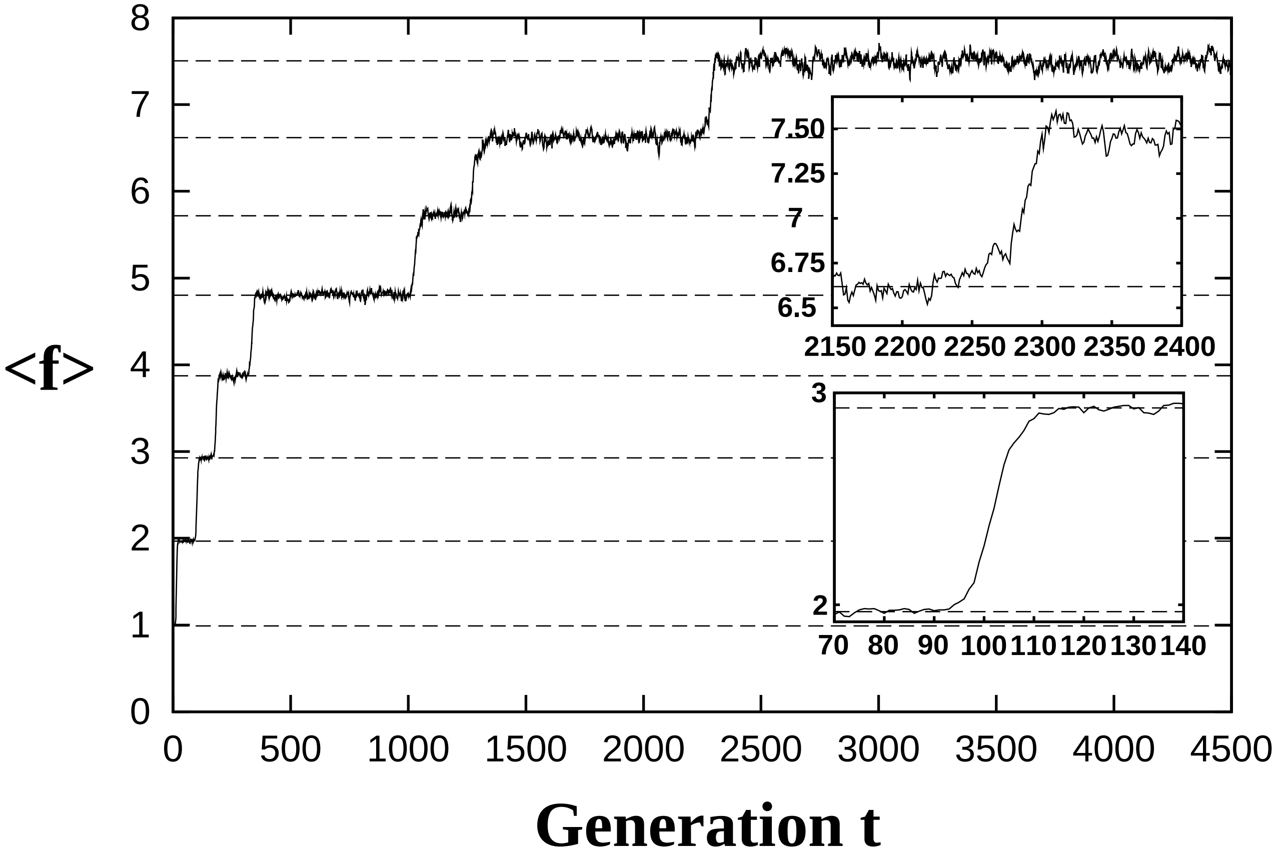

Figure 1: Average fitness in the population over 4500 generations for a GA run with $N = 8$, $K = 8$, $q = 0.001$, and $M = 500$. The horizontal lines are the predicted fitness values for the epochs. The inset plots compare the speed of the innovations at early (lower inset) and late (upper inset) generations. At later generations they are slower: note that the upper inset plot covers a time scale that is more than three times longer than the lower.

Formally, the GA behavior is specified by an iterated procedure, $\mathcal{P}_{t+1} = \mathcal{G}\mathcal{P}_t$, in which the dynamic $\mathcal{G}$ represents the implementation of the selection and mutation steps.

Unlike many GAs this algorithm does not include crossover—the exchange of components from two or more parent strings. Since the main qualitative feature of the dynamics—the occurrence of fitness epochs—is not changed by omitting crossover and the analysis is simplified considerably without it, it is not included in this initial work. Also, this GA is simple in that there is a trivial mapping from the genetic specification to a phenotype: the fitness is evaluated directly on the genome.

Figures 1 and 2 display the results of two runs of this GA with different parameter settings. The average fitness in the population $\langle f \rangle$ over time t (generation number) is shown in both. Figure 2, in addition, shows the best fitness in the population as a function of time. In these

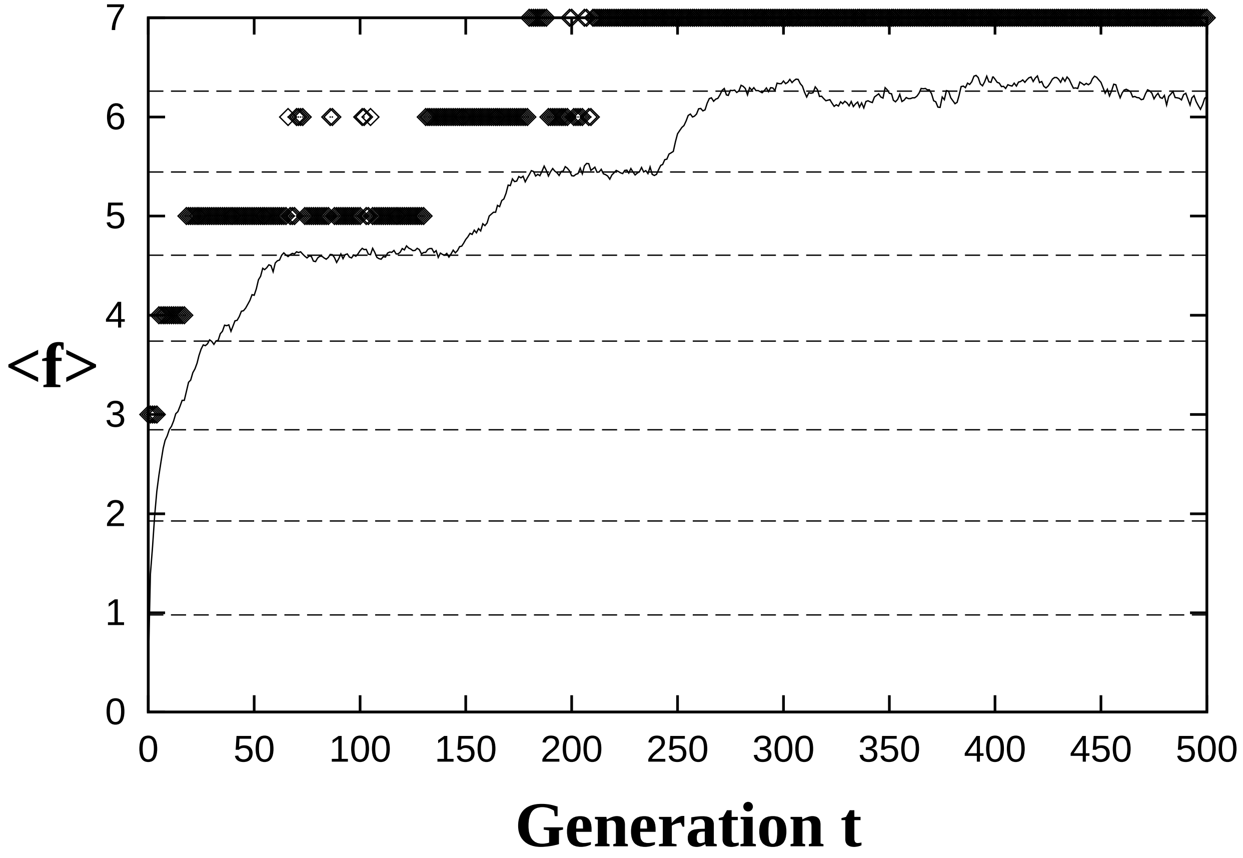

Figure 2: Average fitness (solid line) and best fitness (diamonds) in the population over 500 generations for a GA run with $N = 7$, $K = 4$, $q = 0.004$, and $M = 500$. The horizontal lines are the predicted fitness values for the epochs.

plots the fitness epochs are clear. The larger block size and the lower mutation rate of the run in Figure 1 seem to lead to more sharply defined epochs than in the run of Figure 2. We would like to be able to explain why the fitness follows this pattern of stepwise increase. In particular, a complete analysis of the simplified GA on this class of fitness functions would provide an analytical framework that, given a set of parameters (N, K, q, and M), can answer the following questions: (i) How many fitness epochs can possibly occur? (ii) At what fitness levels do the fitness epochs occur? (iii) What is their expected duration? (iv) What is the expected speed of the innovations between fitness epochs? In this paper we will answer the first two questions and give some preliminary insights into the last two.

We assume that the dynamics of the population $\mathcal{P}_t$ can be captured by analyzing the dynamics of the distribution $\vec{P}(t)$ of fitnesses in the population. In the $M \rightarrow \infty$ limit of infinite population size, the dynamics of $\vec{P}(t)$ will become deterministic and can be solved analytically.

A fitness distribution $\vec{P} = (P_0, \ldots, P_N)$ is a point on the $N+1$ dimensional simplex, where P_f is the proportion of individuals with fitness f, $0 \leq f \leq N$, and the distribution is normalized. Thus, the state space is the simplex $\Lambda_\infty = \{\vec{P} \in \mathbb{R}^{N+1} : \sum_{f=0}^{N} P_f = 1\}$. The average fitness $\langle f \rangle$ in the population is simply $\langle f \rangle = \sum_{f=0}^{N} f P_f$. To obtain the infinite-population dynamics we construct a generation operator $\mathbf{G}$ that incorporates the effects of selection and mutation on the population at the level of the time-dependent fitness distribution; that is, the GA dynamics are now specified by $\vec{P}(t+1) = \mathbf{G}(\vec{P}(t))$, where $\mathbf{G} = \mathbf{M} \circ \mathbf{S}$ is the product of the mutation and selection operators $\mathbf{M}$ and $\mathbf{S}$, respectively.

To construct the mutation operator, we assume that all "incorrect" blocks (those not consisting of all 1s) in the population can be taken to be independent and, furthermore, that all incorrect blocks have not been correct in their past. In other words, strings in the population are not too similar and strings in which blocks have just been destroyed have a low probability of being selected. Under these assumptions, we can consider the dynamics of a string $\mathbf{s}$ consisting of a single block of K bits under mutation alone to obtain an approximation to the probability that mutation will turn an incorrect block into a correct block. At each time step, each bit in $\mathbf{s}$ is mutated with probability q. Starting from a random K-bit string at time $t = 0$, a Markov chain analysis gives the average time $T(q, K)$ until the correct constellation of K 1s appears for the first time. Our calculation shows that the first-passage-time distribution for this problem is exponential to a high approximation. As a result, the probability A that an "incorrect" block will be transformed into a correct block in one time step is constant: $A = 1/T(q, K)$. The probability D that a correct block is transformed into an incorrect block in one time step is $D = 1 - (1 - q)^K$.

Using the single-block expressions for A and D we can determine the probability $\mathbf{M}_{ij}$ that a string $\mathbf{s}$ of length NK with fitness j will be transformed into a string with fitness i in one time step. $\mathbf{M}_{ij}$ is the sum over all the probabilities that k incorrect blocks in the string will become correct in that time step and l correct blocks will become incorrect in that time step, such that $j + k - l = i$. The matrix entries of the mutation operator $\mathbf{M}$ are thus given by

$$\begin{aligned} \mathbf{M}_{ij} = \; & \textstyle\sum_{k=0}^{N-j} \sum_{l=0}^{j} \delta_{j+k,i+l} \binom{N-j}{k} A^k (1-A)^{N-j-k} \\ & \textstyle\binom{j}{l} D^l (1-D)^{j-l} \end{aligned} \tag{1}$$

where $\delta_{i,j}$ is the Kronecker delta function.

To construct the selection operator, recall that our simple GA uses fitness-proportionate selection. This means that the fraction P_f^s of strings with fitness f after selection is proportional to both f and the fraction P_f of strings with fitness f before selection. That is, $P_f^s \propto f \; P_f$. Since $\vec{P}^s$ remains normalized, it can easily be seen that $P_f^s = f \; P_f / \langle f \rangle$, where $\langle f \rangle$ is the mean fitness of the distribution $\vec{P}$. We can therefore write the matrix entries $\mathbf{S}_{ij}$ of the selection operator $\mathbf{S}$ as $\mathbf{S}_{ij} = i \delta_{ij} / \langle f \rangle$. Notice that, in contrast to the mutation operator $\mathbf{M}$, the selection operator $\mathbf{S}$ is nonlinear: $\mathbf{S}$ scales with the inverse of the average fitness $\langle f \rangle$.

The generation operator $\mathbf{G}$ is simply the product of mutation and selection operators $\mathbf{M}$

and $\mathbf{S}$:

$$\mathbf{G}_{ij} = \sum_{k=0}^{N} \mathbf{M}_{ik}\mathbf{S}_{kj}. \tag{2}$$

Note that the fitness distribution in the population at time t is given by the t'th iterate of $\mathbf{G}$:

$$\vec{P}(t) = \mathbf{G}^t[\vec{P}(0)]. \tag{3}$$

To analyze $\mathbf{G}^t$ we will first construct a linearized version $\tilde{\mathbf{G}} = \mathbf{M} \cdot \tilde{\mathbf{S}}$, linearizing the selection operator by setting $\mathbf{S} = \tilde{\mathbf{S}}/\langle f \rangle$. Since $\tilde{\mathbf{G}}$ is proportional to $\mathbf{G}$ at each time step, we can formally write the t'th power of $\mathbf{G}$ as:

$$\mathbf{G}^t[\vec{P}(0)] = C[t, \vec{P}(0)]\, \tilde{\mathbf{G}}^t \cdot \vec{P}(0) \tag{4}$$

where $C[t, \vec{P}(0)]$ is a constant that depends on t and the fitness distribution at time $t = 0$.

Since, in general, $\tilde{\mathbf{G}}$ has $N+1$ distinct eigenvalues f_i and $N+1$ distinct normalized eigenvectors $\vec{V}^i$, its diagonal form is given by $R^{-1} \cdot \tilde{\mathbf{G}} \cdot R$, where $R_{ij} = V_i^j$ is the similarity transformation matrix. The fitness distribution in this basis is

$$\vec{P}(t) = \sum_{i=0}^{N} \alpha_i(t) \vec{V}^i. \tag{5}$$

and we can think of a given $\vec{P}(t)$ as a distribution $\vec{\alpha}(t)$ over the different eigenvectors $\vec{V}^i$. In this basis, equation 3 takes on a very simple form:

$$\alpha_i(t) = C[t, \vec{\alpha}(0)]\, f_i^t \alpha_i(0) = \frac{f_i^t \alpha_i(0)}{\sum_i f_i^t \alpha_i(0)}, \tag{6}$$

where the constant $C[t, \vec{\alpha}(0)]$ is easily determined by demanding that the distribution $\vec{\alpha}(t)$ over the eigenvectors be normalized. Equation 6 shows that the fractions $\alpha_i(t)$ of the non-principal eigenvectors $\vec{V}^i$ eventually all become exponentially damped. In the limit of $t \to \infty$ the fraction α_N of the principal eigenvector $\vec{V}^N$ exponentially approaches 1, [7]. Furthermore, we obtain a simple expression for the average fitness in the population as a function of time:

$$\langle f(t) \rangle = \frac{\sum_i f_i^{t+1} \alpha_i(0)}{\sum_i f_i^t \alpha_i(0)}. \tag{7}$$

Figure 3 shows the predicted infinite-population dynamics for $N = 3$, $K = 4$ and $q = 0.01$ together with the average experimental results over 20 runs of the GA using those parameters and a large population ($M = 10^4$). The error-bars denote plus or minus two standard deviations from the average over the 20 runs. For large populations ($M > 2^{NK}$) the actual dynamics is accurately predicted by the infinite-population dynamics. Notice that for populations of this size the average fitness increases smoothly as a function of time and there are no discernible fitness epochs.

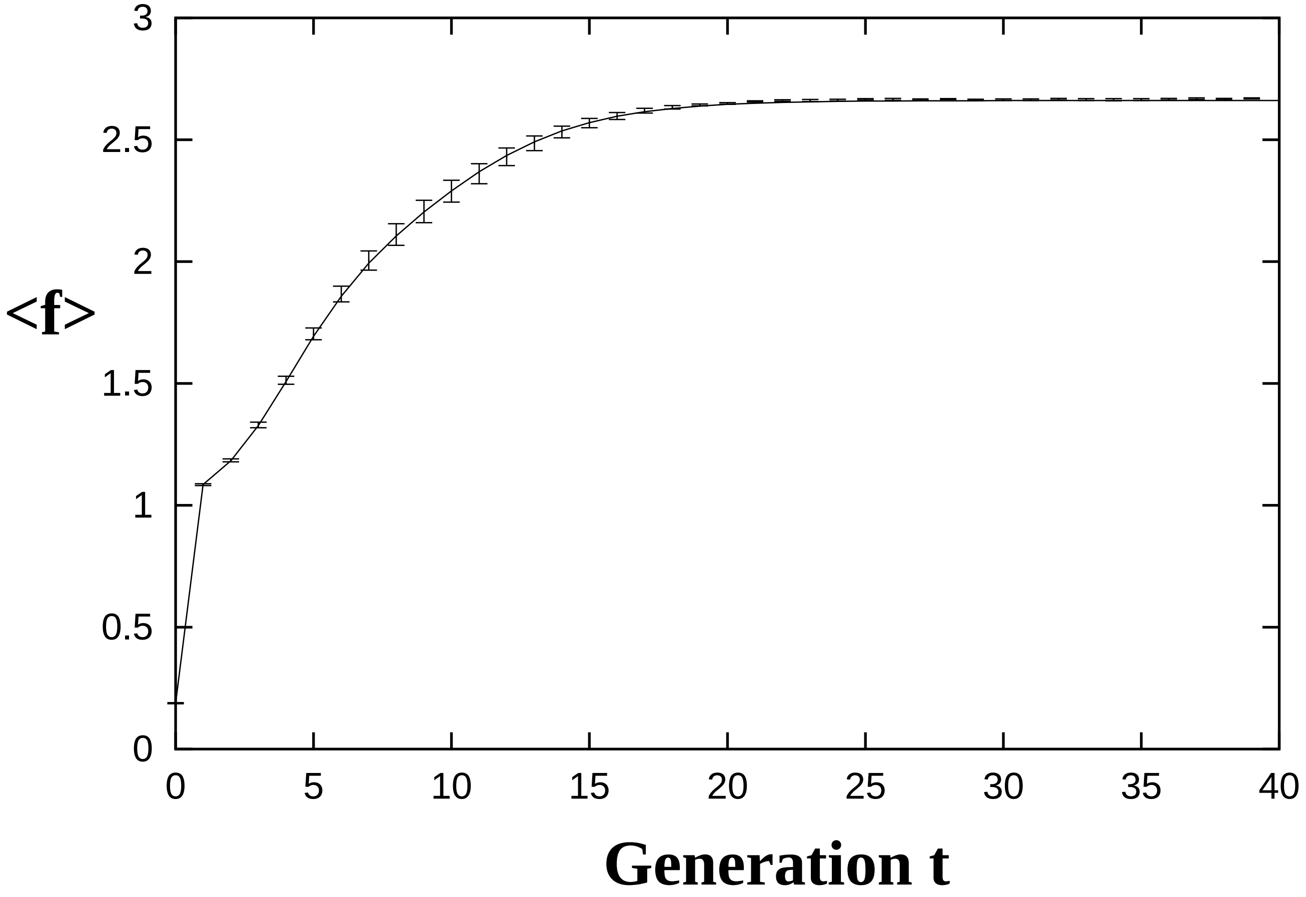

Figure 3: Theoretical prediction of average fitness over time for the parameters $N = 3$, $K = 4$, $q = 0.01$, and $M = 10^4$ (solid line) along with average experimental results over 20 runs with the same parameters. The error-bars give plus and minus two standard deviations from the experimental average.

Several additional properties of the GA's behavior follow directly from our analysis of the infinite-population dynamics given in equation 7. First, for any $\epsilon > 0$ and $0 < \gamma < 1$ and any finite number of time steps t, there is a population size $M_{\epsilon,\gamma}$ such that with probability greater than γ the finite-population trajectory stays within a distance ϵ of the infinite-population trajectory $\vec{P}(t)$ for all t time steps, for population sizes $M > M_{\epsilon,\gamma}$. (The proof of this is analogous to a similar proof in [4].) Second, the average fitness $\langle f(t) \rangle$ is a monotonically increasing function of time. Finally, using Perron's theorem for positive matrices, we can show that the infinite-population dynamics has a unique asymptotic fixed-point fitness distribution $\vec{P}(\infty)$ which is given by $\tilde{\mathbf{G}}$'s principal eigenvector $\vec{V}^N$.

For finite populations of size M, the fitness-distribution state space is a lattice Λ_M over the $N+1$ dimensional simplex Λ_∞ with lattice spacing M^{-1}:

$$\Lambda_M = \{\vec{P} \, : \, P_f = \frac{n_f}{M}, \, n_f \in \mathbb{N}, \, \sum_{f=0}^{N} n_f = M\}, \tag{8}$$

where n_f is the number of individuals with fitness f in the population.

As in the infinite-population case, applying $\mathbf{G}$ to a fitness distribution $\vec{P}$ in Λ_M gives the expected fitness distribution $\langle \vec{P'} \rangle = \mathbf{G}(\vec{P})$ at the next generation. The *actual* fitness distribution $\vec{P'}$ at the next time step will be a random sample of size M of the distribution $\mathbf{G}(\vec{P})$. This induces stochasticity in the finite-population dynamics. In general, for a finite population of size M, the probability $\Pr[\vec{P} \rightarrow \vec{P'}]$ that a fitness distribution $P_f = n_f/M$ will go to a fitness distribution $P'_f = m_f/M$ under mutation and selection will be given by a multinomial distribution with mean $\mathbf{G}(\vec{P})$. That is,

$$\Pr[\vec{P} \rightarrow \vec{P'}] = M! \prod_{f=0}^{N} \frac{[\mathbf{G}_f(\vec{P})]^{m_f}}{m_f!}, \tag{9}$$

where $\mathbf{G}_f(\vec{P})$ is the expected proportion $\langle P'_f \rangle$ of individuals with fitness f at the next generation.

The expected change $\langle \Delta P_f \rangle$ in the fth component of the fitness distribution is given by the difference $\mathbf{G}_f(\vec{P}) - P_f$. If $\langle \Delta P_f \rangle$ is much smaller than the lattice spacing $1/M$, the number of individuals with fitness f most likely will not change. Imagine the case where no individual has a fitness higher than k, i.e. $P_f = 0, f > k$, and the population is confined to the $k+1$ dimensional subsimplex Λ_M^k. If the probability of creating new blocks is small with respect to $1/M$ then the higher components $(f > k)$ of the fitness distribution are likely to remain 0 for some time and the population will stay in Λ_M^k. In other words, the flow $\langle \Delta P_f \rangle$ out of the subsimplex $(f > k)$ is small with respect to the lattice spacing $1/M$.

At the same time, mutation (which continues to destroy blocks) and selection will steer the population to a metastable fixed point within Λ_M^k, where it remains until a new block is created and the associated string spreads in the population. This metastable fixed point is given by the largest eigenvalue of the $k+1$-dimensional submatrix of $\tilde{\mathbf{G}}$. For small mutation rates q it can be shown that the largest eigenvectors of the $k+1$-dimensional submatrices lie close to the eigenvectors $\vec{V}^k$ of the full matrix $\tilde{\mathbf{G}}$. So, although the generation operator $\mathbf{G}$ has only one fixed point in Λ_∞, it's nonprincipal eigenvectors $\vec{V}^k$ lie very close to the flow's metastable fixed points in the lower dimensional subsimplices Λ_M^k, around which the population stabilizes until strings with fitness $f > k$ are discovered. In short, the combination of the state-space discreteness and the stochasticity of the GA dynamics—both induced by the population's finite size—causes populations moving in Λ_M^k to temporarily stabilize within these subsimplices. In this way, metastability is induced via finite populations.

Using the fact that the mutation operator $\mathbf{M}$ is a stochastic matrix, we can show that each eigenvalue f_i of $\tilde{\mathbf{G}}$ is equal to the average fitness $\langle f \rangle$ in the population for the fitness distributions of the corresponding eigenvector $\vec{V}^i$. We therefore expect the fitness epochs to occur at fitness levels that simply correspond to the eigenvalues of $\tilde{\mathbf{G}}$. Using a second order perturbation analysis in q of the eigenvalues of $\tilde{\mathbf{G}}$, the fitness levels are approximated, at low mutation, by

$$f_i(q) \ = i - [i^2 K + i(N-i)A_1]q + \mathcal{O}(q^2) \tag{10}$$

where $A_1 q$ is the probability to align a block to first order in q—that is, $A = A_1 q + \mathcal{O}(q^2)$—and where $A_1^{-1} = 2^{-K} \sum_{j=1}^{K} j^{-1} \binom{K}{j}^{-1} \left(\sum_{i=0}^{j-1} \binom{K}{i} \right)^2$.

In addition to the experimental data from the GA simulation runs, Figures 1 and 2 show the fitness values corresponding to the eigenvalues f_i of $\tilde{\mathbf{G}}$, which were calculated numerically, as horizontal dashed lines. It is clear that the eigenvalues accurately predict the fitness levels at which the epochs occur.

Figures 1 and 2 show that epoch duration varies among the different epochs and that not all possible fitness epochs occur in any particular run of the genetic algorithm. Moreover, we observe that on average the higher fitness epochs have longer durations and are more likely to be visited in a given run.

These observations can be understood by a linear stability analysis of $\mathbf{G}$ around the metastable states $\vec{V}^k$. Solving for the spectrum of the Jacobian matrix $\mathbf{DG}$, we find that the eigenvalues λ_i^k of the Jacobian matrix at $\vec{V}^k$ are given by:

$$\lambda_i^k = \frac{f_i(q)}{f_k(q)}, (k \neq i) \tag{11}$$

From this, we see that the number of stable dimensions of a metastable state $\vec{V}^k$ is equal to k (where $0 \leq k \leq N$). Since the actual center of the fitness epoch, given by the largest eigenvalue of the $k+1$-dimensional submatrix of $\tilde{\mathbf{G}}$, is slightly offset from $\vec{V}^k$ in the direction of the unstable dimensions, one sees that qualitatively the duration of an epoch is inversely proportional to the number of unstable dimensions and the size of the eigenvalues λ_i^k of the unstable dimensions ($i > k$). For larger k there are fewer unstable dimensions and the sizes of the λ_i^k are relatively smaller. Therefore, we expect the higher-fitness epochs to have a longer duration on average. The probability that the population will visit a particular epoch is determined by the probability that the population will fall onto one of the attracting dimensions of that epoch in the course of its evolution. Again qualitatively, we expect that an epoch with a larger number of stable dimensions will therefore also have a higher probability of being visited in any particular run.

We see from equations 10 and 11 that for all unstable dimensions ($\lambda_i^k > 1$) of a metastable state $\vec{V}^k$ we have $\lambda_i^k \geq 1 + 1/k$ as $q \to 0$. For small values of k ($= \mathcal{O}(1)$) the eigenvalues corresponding to the unstable dimensions of $\vec{V}^k$ are therefore considerably larger than 1. This means that as soon as the fitness distribution acquires a minimal component of size $1/M$ or more in an unstable dimension, it will start moving away from the metastable state exponentially fast. This explains the occurrence of the steep innovations between the epochs. Once an unstable dimension directed towards a higher metastable state is discovered by evolution, the fitness distribution will move to this new metastable state exponentially fast. For higher mutation rates, the range of the ratios f_i/f_k about 1 decreases, and we expect the innovations to become less steep. We indeed see more gradual innovations in the run with the higher mutation rate (Figure 2). The eigenvalues f_i/f_k of the unstable dimensions ($i > k$) approach 1 from above for increasing values of k. We therefore expect the innovations between higher-fitness epochs to be slower than the innovations between lower-fitness epochs. This effect is enhanced by the fact that the first-order term i^2N increases with i. The inset plots in Figure 1 show that innovations at later generations are indeed slower. The time scale for the upper plot (later generations) is three times that of the lower (earlier generations). Finally, since there are fewer stable dimensions for smaller k and the stable dimension eigenvalues λ_i^k are smaller as well, we expect the fitness fluctuations in the lower-fitness epochs to be smaller than the fluctuations in the higher-fitness epochs. The plots also illustrate this phenomenon.

To summarize and conclude: evolutionary search algorithms are stochastic dynamical systems in which a large set of identical elements evolve in parallel and under the influence of one another through a state space. Macroscopic states for these systems are often defined in terms of the first moments of the distributions over the state variables of the elements. A commonly observed qualitative behavior is that the mean of some state variable alternates between periods of stasis and sudden change. In this paper we analyzed a simplified genetic algorithm as an example of such a system and derived macroscopic equations of motion, from the microscopic dynamics specified by the GA, that explain the occurrence of these periods of stasis and sudden change.

We constructed an operator $\mathbf{G}$ that describes the GA's deterministic dynamics in the limit of infinite populations. By diagonalizing the linearized version $\tilde{\mathbf{G}}$ we were able to obtain closed-form expressions for the evolution of the population fitness distribution. In the finite-population case the dynamics was still governed by the operator $\mathbf{G}$. Due to the finite population, however, the state space became discrete and the finite-population sampling noise caused the dynamics to become stochastic. This, in turn, led to the appearance of metastable fitness distributions in the vicinity of the hyperbolic fixed points of $\mathbf{G}$. We were able to identify the fitness epochs in the dynamics with these hyperbolic fixed points. This gave us both the fitness levels at which the epochs occur and the associated metastable fitness distributions. We also explained several other dynamical features, such as the innovations' short durations and the appearance and disappearance of epochs.

The behavior of evolutionary search algorithms is often informally described as moving along a "fitness landscape" directly defined by a fitness function; cf. [1]. It is clear from our experiments and our analysis of the underlying mechanisms that this geographic metaphor is misleading. First, the fitness function is only a partial determinant of the search dynamics. Population size, mutation rate, and other parameters can radically alter the time-dependent metastable behavior—revealing, hiding, or even inducing much of the structure in the fitness landscape. Note, for instance, that our fitness function is simple: it has a single peak, but otherwise no local optima, and it is not "rugged". Second, our analysis suggests that by combining the fitness landscape with the population dynamics—as we did in constructing **G**—one can obtain an effective, but different, "landscape" that guides the evolutionary search.

This work was supported at the Santa Fe Institute by NSF IRI-9320200, DOE DE-FG03-94ER25231, and ONR N00014-95-1-0975, and at UC Berkeley under ONR N00014-95-1-0524.

References

[1] T. Bäck, *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*, Oxford University Press, Oxford (1996); D. B. Fogel, *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, IEEE Press, New York (1995); D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA (1989); Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Artificial Intelligence Series. Springer-Verlag (1992); M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, Cambridge, MA (1996).

[2] S. Wright, Evolution **36** (1982) 427-443; A. Bergman and M. W. Feldman, "Question marks about the period of punctuation", Santa Fe Institute Working Paper 96-02-006 (1996); M. Huynen, P. F. Stadler, and W. Fontana, Proc. Natl. Acad. Sci. **93** (1996) 397-401; K. Lindgren, in C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, Addison-Wesley, Reading, MA (1992) 295-312; J. P. Crutchfield and M. Mitchell, Proc. Natl. Acad. Sci. **92** (1995) 10742-10746; M. Mitchell, J. P. Crutchfield and P. T. Hraber, Physica D **75** (1994) 361-391; T. S. Ray, in C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, Addison-Wesley, Reading, MA (1992) 371-408.

[3] M. D. Vose and G. E. Liepins, Complex Systems **5** (1991) 31-44.

[4] A. E. Nix and M. D. Vose, Ann. Math. Art. Intell. **5** (1991) 79-88.

[5] A. Prügel-Bennett and J. L. Shapiro, Phys. Rev. Lett. **72** (1994) 1305-1309; A. Prügel-Bennett and J. L. Shapiro, Physica D (1996) in press; M. Rattray and J. L. Shapiro, J. of Phys. A **29** (1996) 7451-7473.

[6] S. Forrest and M. Mitchell, in L. D. Whitley, editor, *Foundations of Genetic Algorithms 2*, Morgan Kauffman, San Mateo, CA (1993) 109-126; M. Mitchell, J. H. Holland, and S. Forrest, in *Advances in Neural Information Processing Systems 6*, Morgan Kaufmann, San Mateo, CA (1994) 51-58.

[7] Our equation 3, for infinite populations, is analogous to a discrete-time version of the differential equations of the Eigen model of molecular evolution: M. Eigen, Naturwissenschaften **58** (1971) 465-522; M. Eigen and P. Schuster, Naturwissenschaften **64** (1977) 541-565. The asymptotically dominant eigenvector $\vec{V}^N$ of our model is analogous to a so-called "quasi-species" of molecules: an asymptotically stable distribution over different molecular species. Our analysis shows that the evolutionary dynamics for finite populations can get trapped in regions close to the nonprincipal eigenvectors $\vec{V}^i$ of the generation operator. This suggests that in practical instances of molecular evolution, where the population size is typically much smaller than the size of the sequence space, "metastable quasi-species" are to be expected in the vicinity of the nonprincipal eigenvectors $\vec{V}^i$ of the generation operator (or transfer-matrix, as it has been called in reference to the Eigen model: I. Leuthäusser, J. of Stat. Phys. **48** (1987) 343-360).