WHAT ARE GENETIC ALGORITHMS? A MATHEMATICAL PRESPECTIVE*

MICHAEL D. VOSE†

Abstract. This talk presents a "big picture" view of genetic search in a general, abstract setting. It limits consideration to simple generational versions of time invariant Markovian GAs (the next generation depends only upon the current population) with the aim of uncovering, in general terms, their inherent emergent behavior. Along the way, a few issues related to classical "GA theory" are touched upon so as to sketch the context out of which the material presented in the remainder of the talk grew, and to indicate a few of the problems it partially addresses.

1. Introduction. This talk concerns simple generational versions of time invariant Markovian GAs (the next generation depends only upon the current population). The introduction briefly touches upon a very few of the many parts of the mosaic that, historically, has been referred to as "GA threory", so as to provide context and contrast for the remainder of the talk. There results are presented which were motivated by, and partially respond to, shortcomings of the classical theory. It is assumed throughout that the audience is acquainted with the jargon commonly used in the field.

2. Intrinsic parallelism and no free lunch. Beginning with John Holland (if not before) in his book "adaptation in natural and artificial systems" (1975), the concept of schema was introduced as a mechanism whereby genetic search could be analyzed and understood. Although sets of elements, as opposed to individuals, were emphasized, Holland consistently stressed the importance of creating *new* samples (elements not in the current or past populations) throughout his development. To summarize the central ideas:

- Collections of elements (schemata) in the population gain representation in rough correspondence to their average fitness.
- An above average schema increases its share of trials exponentially.
- Genetic search rapidly explores sets of schemata of above average performance which can be produced from each other by relatively few crossovers, while not significantly slowing the overall search for better optima.

Presented in harmony with this schema-based view was the conjecture that by virtue of an element simultaneously belonging to a plethora of schemata (which is what *intrinsic parallelism* refers to), a tremendous flow of information is available to the genetic paradigm by which it gains tremendous power.

*This research was supported by the National Science Foundation IRI-9224917.

†C.S. Dept., 107 Ayres Hall, The University of Tennessee, Knoxville, TN 37996-1301. Email: vose@cs.utk.edu

This latter point – power flowing from intrinsic parallelism – has been laid to rest by the "no free lunch" theorem (Wolpert, D. and W. Macready, 1994). The NFL (no free lunch) theorem, roughly speaking, asserts that all black box search (BBS) strategies have similar performance when considered over all test functions[1]. The basic principle is straightforward. Suppose it is not a priori known what value will be returned by the objective function; this may be regarded as uncertainty as to what function g is being optimized. By defining the value to be returned – in an arbitrary way – when evaluating at a new point in the domain, one can arrange for any behavior to be manifest while incrementally defining the objective function as the search progresses. When all points in the finite domain have been explored, the objective function will have thereby been defined, and no uncertainty remains as to which function g was to be optimized.

Since the values returned as described above may be arbitrarily chosen, a GA can be made – through suitable choices – to perform either better or worse than any (other) BBS strategy applied to a given (typical) function f.[2] Of course, such performance might be manifest only when the GA is applied to an objective function g which differs from f; the incremental definition as described above need not result in the determination of f as the function which the GA was optimizing. In its full generality, the NFL theorem has the consequence that, on average, the effectiveness of genetic search is no better than that achieved by enumeration: all BBS strategies have equivalent performance; they merely exhibit it on different functions.[3]

3. Building blocks and the schema theorem. The belief in the power of intrinsic parallelism spawned attendant conjectures, like, for instance, the notion that mutation is an inferior search operator, and the idea that high cardinality alphabets yield inferior results. For a period of time, perspectives assuming the central importance of the *number* of schemata which survived, or that were processed, or which simply existed (given some particular representation), became, for better or for worse, dominant.

Popularized by David Goldberg in his book "Genetic Algorithms, in Search, Optimization, & Machine Learning" (1989), the *building block hypotheses* is the assertion that highly fit, short, low order schemata are recombined ("processed" by crossover) to lead to better performance (i.e., to create new, more highly fit elements). By the light the NFL theorem sheds, a GA will work well on some problems, poorly on others, and on average no better than enumeration. Under that light, the building block hypotheses might best be interpreted as a conjecture as to *what kinds* of functions a GA would perform well on: those for which new and better

[1] from a finite domain $\mathcal{X}$ to a finite codomain $\mathcal{Y}$.

[2] Provided performance is measured with respect to the sequence of values encountered during search, ignoring repeats.

[3] There are, naturally, conditions and technicalities; they do not, however, detract from the negative conclusions about the power of intrinsic parallelism.

elements will be produced, by crossover, from highly fit, short, low order schemata.

This leads naturally to the question: What new elements *are* produced by a genetic algorithm? The *schema theorem* is an inequality giving a lower bound on the expected number of instances of schemata in the next generation. The lower bound is zero, however, for all schemata not already present in the current population. A little thought will reveal that *which* elements outside of the current population become members of the next generation is crucial to the course of genetic search. To worry about *how many* within the current population survive, are processed, or exist, is to miss an important point.

A devastating consequence follows from the fact that the schema theorem does not nontrivially address what proportions, of which strings, schemata are expected to be comprised. Without such knowledge, schemata utilities become unknown in the next generation. Consequently, the schema theorem's ability to predict – even about strings within the current population – evaporates, in general, after a single generation.

Apparently neglected for some period of time, a paper by Bridges and Goldberg ("An analysis of reproduction and crossover in a binary-coded genetic algorithm") remedied that situation in 1987. A formula was presented which gave the expected representation of *every* schema in the next generation. Although limited to proportional selection, one point crossover, and zero mutation, their result was progress, in that an inequality had been sharpened to an equality, and the focus on "how many schemata?" had been broadened – in terms of proven analytical results – to include the question "*which new schemata?*".

4. Sampling error and facetwise analysis. Facetwise analysis (ignoring or eliminating problem aspects; treating the details of their interaction as inconsequential) emerged as a popular method to reason (albeit heuristicly) in the face of complexity and uncertainty. This practice was fostered by the reality that genetic algorithms are stochastic and nonlinear.

In circumstances where the building block hypothesis was thought appropriate, the idea arose that the GA's inherent mechanism could be an internal source of *sampling error*, potentially interfering with anticipated outcomes. Selection, for example, might choose unrepresentative parents, mutation or crossover could produce unrepresentative offspring. From the schemata perspective, the population size might be too small, setting the stage for an unrepresentative estimate of schemata utilities. Considerations like these spawned alternative operators less prone to sampling error, and also influenced theories of population sizing. Taking the limit, as population size goes to infinity, ameliorates these concerns, however. The resulting dynamics are deterministic and equivalent to that induced by the function which produces the expected next generation.

In the field of genetic algorithms, this methodology was initiated by

Holland and carried forward by his students. For example, the central ideas summarized in section 2, as well as the schema theorem, are statements about expectations or are justified by an implicit appeal to a trajectory of expectations (as, for example, the exponential increase in above average schemata). Not to suggest anything amiss with facetwise analysis, it is nevertheless fair to say that for a period of time the description of what, precisely, it established, had too frequently been left to the imagination or treated in a cavalier manner. The following subsections illustrate *fallacious* conclusions based on interpretations of "results" of "GA theory" which too commonly are made.

5. Static schema analysis. Consider the objective function $\{(00, 2.7), (01, 1.0), (10, 1.0), (11, 1.1)\}$ and assume the crossover rate χ is 0.6. The nontrivial schemata, with their associated fitness, are

$$\begin{aligned} 0{*} &\longleftrightarrow 1.85 \\ {*}0 &\longleftrightarrow 1.85 \\ 1{*} &\longleftrightarrow 1.05 \\ {*}1 &\longleftrightarrow 1.05 \end{aligned}$$

The optimal element is over 145% more fit than any other, and the schemata containing it ($0*$ and $*0$) are over 76% more fit than their competitors; there is absolutely no deception (the schemata containing 00 win every "competition"). Schemata gain representation in correspondence to their fitness, and above average schemata increase exponentially; thus $0*$ and $*0$ – if initially present in the population – will quickly come to dominate. These schemata cannot be disrupted by crossover (they are too short), so in the absence of mutation, the exploratory operators cannot interfere with the exploitation of these building blocks. Sampling error is a potential problem, but that can be tamed by choosing population size sufficiently large.

It can safely be concluded in this case that given a zero mutation rate and an initial population containing fixed nonzero proportions of every element, a GA will with high probability converge to the optimal, provided the population size is not too small.

The conclusion of the previous paragraph sounds reasonable. It is false however. In fact, there exists a function h of the crossover rate χ which increases to infinity as χ increases to 1.0, such that the previous paragraph is false with respect to the function $\{(00, h(\chi)), (01, 1.0), (10, 1.0), (11, 1.1)\}$.

It is interesting to note, for this example, what the problem is *not*. The computation of schema utilities with respect to a uniform population (equal representation of strings), instead of with respect to the initial population from which convergence to a suboptimal is likely, is *not* to blame. Initial populations exist for which the initial conditions hold with respect to nonuniform schema utilities[4], yet convergence to a suboptimal remains

[4] The optimal element remains over 145% more fit than any other, and the schemata

the likely event (even though fixed nonzero proportions of every element are initially present and the population size may be arbitrarily large). The principal behind the example is the fact that the schema theorem's ability to predict – even about strings within the current population – evaporates, in general, after a single generation.

6. Neglecting finite population effects. Because taking the limit, as population size goes to infinity, yields a deterministic system whose dynamics coincide with that induced by the function which produces the expected next generation, the infinite population model can be used to determine the expected behavior of a genetic algorithm.

The conclusion of the previous paragraph is slippery, because the statement is so vague. Following are specific interpretations, each of which is a *false* assertion.

- The path followed by the infinite population model is the expected path followed by a GA.
- If the infinite population model converges to a population q, then q is representative of the GA's steady state distribution (as a Markov chain).
- If the infinite population model indicates that certain elements are likely to emerge in early generations, then it is probable for these elements to likewise emerge during genetic search.

This list of errors is not exhaustive, but these are not uncommon misconceptions.

Equally illusory is the ignis fatuus that because such notions are false, the infinite population model is not of fundamental importance to the theory of genetic search.

7. Convergence proofs. Genetic algorithms have been touted as robust optimization methods, similar to simulated annealing in some sense. There are differences of course, one of which being that whereas simulated annealing has a global convergence theory, it is arguably the case that an unimpeachable convergence theory for the simple genetic algorithm does not exist.

Even though there are a variety of "convergence results", those which apply to a simple GA without requiring its alteration amount to little more than the following when distilled to their essence:

> Visit each state, remembering the best state encountered. Therefore, as search progresses and every state is eventually encountered, the best state so far visited will converge to the best state which could be visited – the global optimum.

containing it (0* and *0) remain over 76% more fit than their competitors when nonuniform utilities are computed with respect to the initial population.

Unless severe restrictions are placed on the objective function, this virtual tautology, suitably dressed in the language of Markov chains, is the current state of the art – at least in terms of proven analytical results. To be fair, the convergence theory for simulated annealing is, from a practical applied perspective, hardly better.

In light of the no free lunch theorem, this may very well be as it should be. But the NFL theorem only addresses the sequence of values encountered during search, ignoring repeats. It does not prohibit a theory which could relate observed behavior to mathematical objects determined by the search strategy and the objective function. A nontrivial convergence theory is beginning to appear in this direction, though it speaks to inherent emergent behavior and not to function optimization.

8. Framework. As intimated in the introduction, the majority of theoretical work on genetic search has been facetwise (which, again, is not to suggest anything amiss with the practice of facetwise analysis). It was not until recently that more holistic attempts to deal with the complexities of genetic search were initiated in an attempt to address the shortcomings of incomplete information. As will become apparent, schemata are left behind, playing no part in the development. This is not to suggest that they have no role to play, but their natural place in the general landscape has yet to be discovered.

The view presented is general and abstract, dealing with a class of black box search strategies referred to as Random Heuristic Search (RHS). The simple genetic algorithm is a *special case* of the algorithms discussed. The advantage of this approach is to focus on basic principles by which inherent emergent behavior can be initially approached before delving into specific details. Even so, the analysis proceeds from simple objects encoding *complete* information, rather than a starting point based on preconceived (traditional) designs or assumptions about what may safely be ignored.

9. Representation. Random heuristic search can be thought of as an initial collection of elements P_0 chosen from some search space Ω of finite cardinality n together with some *transition rule* τ which from P_i will produce another collection P_{i+1}. In general, τ will be iterated to produce a sequence of collections

$$P_0 \xrightarrow{\tau} P_1 \xrightarrow{\tau} P_2 \xrightarrow{\tau} \dots$$

The beginning collection P_0 is referred to as the *initial population*, the first population (or *generation*) is P_1, the second population (or generation) is P_2, and so on. Populations are multisets.

Not all transition rules are allowed. Obtaining a good representation for populations is a first step towards characterizing admissible τ. Define the *simplex* to be the set

$$\Lambda \;=\; \{<x_0, ..., x_{n-1}> : \mathbf{1}^T x = 1,\ x_j \geq 0\}$$

where angle brackets $< \cdots >$ denote a tuple which is to be regarded as a column vector and $\mathbf{1}$ denotes the column vector of all 1 s. An element p of Λ corresponds to a population according to the following rule for defining its components

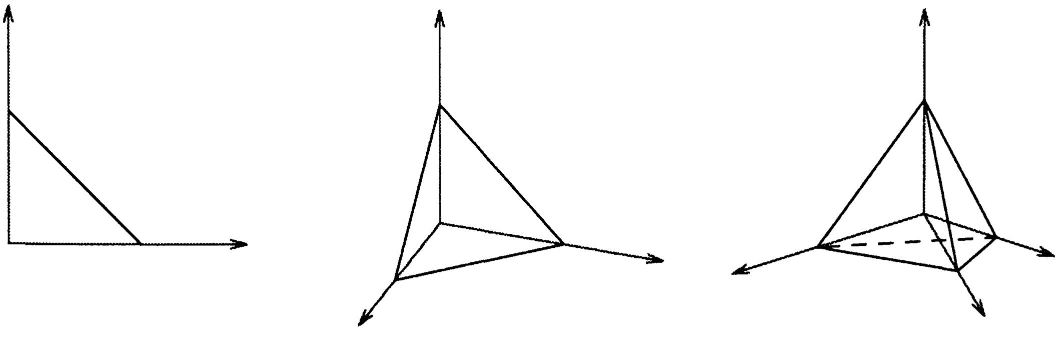
$$p_j \;=\; \text{the proportion in the population of the } j \text{ th element of } \Omega$$

For example, if $\Omega = \{0,1,2,3,4,5\}$ then $n = 6$. The population $\{1,0,3,1,1,3,2,2,4,0\}$ is represented by $p = < .2,.3,.2,.2,.1,.0 >$, given that

coordinate	corresponding element of Ω	percentage of P_0
p_0	0	2/10
p_1	1	3/10
p_2	2	2/10
p_3	3	2/10
p_4	4	1/10
p_5	5	0/10

The cardinality of each generation $P_0, P_1, \ldots$ is a parameter r called the *population size.* Hence the proportional representation given by p unambiguously determines a population once r is known. The vector p is referred to as a *population vector.* The distinction between population and population vector will often be blurred, because the population size is usually fixed. In particular, τ may be thought of as mapping the current population vector to the next.

To get a feel for the geometry of the representation space, Λ is shown in the following sequence of diagrams for n equals 2, 3, and 4. The figures represent Λ (a line segment, triangle, and solid tetrahedron). The arrows show the coordinate axes of the ambient space (the projection of the coordinate axes are being viewed in the second figure, which is three dimensional, and in the last figure where the ambient space is four dimensional).



In general, Λ is a tetrahedron of dimension $n - 1$ contained in an ambient space of dimension n. Note that each vertex of Λ corresponds to a unit basis vector of the ambient space; Λ is their convex hull. For example, the vertices of the solid tetrahedron (right most figure) are at the basis vectors $<1,0,0,0>$, $<0,1,0,0>$, $<0,0,1,0>$, and $<0,0,0,1>$. Assuming that Ω is the ordered set $\{0,1,2,3\}$, they correspond (respectively) to the following populations: r copies of 0, r copies of 1, r copies of 2, and r copies of 3. The center diagram will later be used as a schematic for general Λ, representing it for arbitrary n.

It should be realized that not every point of Λ corresponds to a finite population. In fact, only those rational points expressible with common denominator r correspond to populations of size r. They are the intersection of a rectangular lattice of spacing $\frac{1}{r}$ with Λ:

$$\frac{1}{r} X_n^r \;=\; \frac{1}{r}\{<x_0,\ldots,x_{n-1}>:\; x_j \in \mathcal{Z},\; x_j \geq 0,\; \mathbf{1}^T x = r\}$$

where $\mathcal{Z}$ denotes the set of integers. As $r \to \infty$, these rational points become dense in Λ. The next theorem makes this precise. Since, without a priori knowledge of r, a rational point may represent arbitrarily large populations, a point p of Λ carries little information concerning population size. A natural view is therefore that Λ corresponds to populations of indeterminate size. This is but one of several useful interpretations. Another is that Λ corresponds to sampling distributions over Ω: since the components of p are non negative and sum to 1, p may be viewed as indicating that i is sampled with probability p_i.

THEOREM 9.1. *Let $p \in \Lambda$ denote an arbitrary population vector for population size r, and let ξ denote an arbitrary element of Λ. Then*

$$\sup_{\xi} \inf_{p} \|\xi - p\| \;=\; O(1/\sqrt{r})$$

where the constant (in the "big oh") is independent of the dimension of Λ.

In summary, random heuristic search appears to be a *discrete dynamical system* on Λ through the identification of populations with population vectors. That is, there is some transition rule

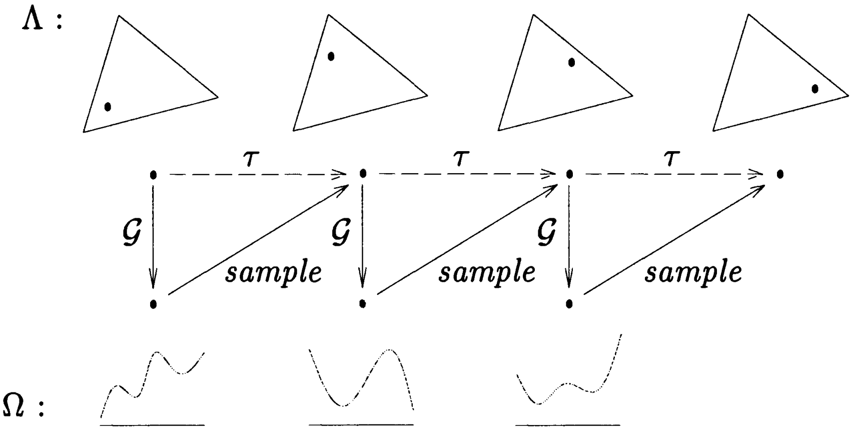
$$\tau \;:\; \Lambda \longrightarrow \Lambda$$

and what is of interest is the sequence of iterates beginning from some initial population vector p

$$p,\quad \tau(p),\quad \tau^2(p),\quad \ldots$$

This view is incomplete however, because the transitions are in general nondeterministic and not all transition rules are allowed. In the next section the stochastic nature of τ will be explained and admissible τ will be characterized.

10. Nondeterminism. Given the current population vector p, the next population vector $\tau(p)$ cannot be predicted with certainty because τ is stochastic. It is most conveniently thought of as resulting from r independent, identically distributed random choices. Let $\mathcal{G} : \Lambda \to \Lambda$ be a *heuristic function* (heuristic for short) which given the current population p produces a vector whose ith component is the probability that the ith element of Ω is chosen (with replacement). That is, $\mathcal{G}(p)$ is that probability vector which specifies the sampling distribution by which the aggregate of r choices forms the next generation. A transition rule τ is admissible if it corresponds to a heuristic function $\mathcal{G}$ in this way. The following diagram depicts the relationship between p, Λ, Ω, $\mathcal{G}$, and τ through a sequence of generations (the illustration does not correspond literally to any particular case, it depicts how transitions between generations take place in general):



The triangles along the top row represent Λ, one for each of four generations. Each Λ contains a dot representing a population. These same populations are also represented in the second row with dots; τ maps from one to the next. The transition arrow for τ is dashed to indicate that it is an induced map, computed by following the solid arrows. The lower row of dots represent images of populations under $\mathcal{G}$. Below each is a curve, suggesting the sampling distribution over Ω which it represents. The line segments in the bottom row represent Ω.

The transition from one generation (upper dot) to the next proceeds as follows. First $\mathcal{G}$ is applied to produce a vector (lower dot) which represents a sampling distribution (curve) over Ω (line segment). Next, r independent samples with replacement (represented in the diagram by *"sample"*) are made from Ω according to this distribution to produce the next generation.

For example, let $\Omega = \{0, 1, 2, 3\}$ and suppose the heuristic is

$$\mathcal{G}(p) \quad = \quad <0, p_1, 2p_2, 3p_3> / \sum ip_i$$

Let the initial population be $p = < .25, .25, .25, .25 >$. Because $\mathcal{G}(p) = < 0, 1/6, 1/3, 1/2 >$, the probability of sampling 0 is 0, of sampling 1 is 1/6, of sampling 2 is 1/3, and of sampling 3 is 1/2. With population size $r = 100$, the transition rule corresponds to making 100 independent samples, with replacement, according to these probabilities.

A plausible next generation is therefore $\tau(p) = <0, .17, .33, .50>$. Note that the sampling distribution $\mathcal{G}(p)$ used in forming the next generation $\tau(p)$ depends on the current population p. Going one generation further, the new population is $\tau(p)$ and the sampling distribution for producing the next generation is given by $\mathcal{G}(\tau(p)) \approx < 0, .07, .28, .64 >$. It is therefore plausible that the second generation could be $\tau^2(p) = <0, .07, .28, .65>$.

Note the conceptually dual interpretation of Λ. It serves as both the space of populations and as the space of probability distributions over Ω. The first natural question is: What is the expected next generation?

THEOREM 10.1. *Let p be the current population vector. The expected next population vector is* $\mathcal{G}(p)$.

A more specific question is: Given current population p, what is the probability that the next generation is q? Let $\theta \in [0, 1]$ be defined by the following form of Sterling's theorem (for $x \in \mathcal{Z}^+$)

$$x! \;=\; \left(\frac{x}{e}\right)^x \sqrt{2\pi x}\, \exp\left\{\frac{1}{12x+\theta}\right\}$$

The function $\theta(x)$ appears in the next theorem.

THEOREM 10.2. *Let p be the current population vector. The probability that* $q \in \frac{1}{r} X_n^r$ *is the next population vector is*

$$r! \prod \frac{(\mathcal{G}(p)_j)^{rq_j}}{(rq_j)!} =$$
$$\exp\left\{-r \sum q_j \ln \frac{q_j}{\mathcal{G}(p)_j} - \sum\left(\ln\sqrt{2\pi r q_j} + \frac{1}{12rq_j + \theta(rq_j)}\right) + O(\ln r)\right\}$$

where summation is restricted to indices for which $q_j > 0$ *and where r is the population size.*

Theorem 10.2 provides qualitative information concerning probable next generations. The expression

$$\sum q_j \ln \frac{q_j}{\mathcal{G}(p)_j}$$

is the *discrepancy* of q with respect to $\mathcal{G}(p)$ and is a measure of how far q is from the expected next population $\mathcal{G}(p)$. Discrepancy is nonnegative and is zero only when q is the expected next population. Hence the factor

$$\exp\left\{-r \sum q_j \ln \frac{q_j}{\mathcal{G}(p)_j}\right\}$$

occurring on the right hand side of theorem 10.2 indicates the probability that q is the next generation decays exponentially (with constant $-r$) as the discrepancy between q and the expected next population increases.

12. Summary. As should be now apparent, the infinite population model – i.e., the expected next generation operator $\mathcal{G}$ – is of fundamental importance to the theory of random heuristic search in general, and to simple genetic search in particular (since it is a special case of RHS).

First, it is identical to the heuristic function which specifies the sampling distribution governing the formation of the next generation. In other words, it repairs the deficiencies inherent in the schema theorem by giving *complete* information about the probabilities with which elements occur in the next generation.

Second, it emerges as the defining component of the transition matrix by which RHS is expressed as a Markov chain. Anything that could ever be proved about simple genetic search therefore corresponds to some property of $\mathcal{G}$, which argues for the investigation of $\mathcal{G}$ as a mathematical object. Abstractly, the heuristic corresponding to the simple GA may be determined by considering the following procedure. This procedure serves as the *definition* of what the terms "simple genetic search" and "simple genetic algorithm" (SGA) refer to:

1. Generate a random population x containing r fixed length binary strings.
2. Choose, with replacement, parents u and v from x (by any fixed selection scheme).
   - Cross u and v (by any fixed crossover rate and type) to produce children u' and v'.
   - Mutate u' and v' (by any fixed mutation rate and type) to produce u'' and v''.
   - Keep, with uniform probability, one of u'' and v'' for the next generation.
3. If the next generation is incomplete, repeat step 2.
4. Replace x by the new generation just formed and go to step 2.

The corresponding heuristic is abstractly defined by

$$\mathcal{G}_i(x) = \Pr\{i \text{ is the result of step 2 } \mid x \text{ is the current population}\}$$

Concretely, $\mathcal{G}$ is explicitly known for a fairly wide range of operators (proportional, ranking, or tournament selection used with arbitrary mutation masks and arbitrary crossover masks). Moreover, a wealth of information is beginning to emerge as the role played by the Walsh transform in the theory of the SGA's heuristic is becoming clear. In this theory, the Walsh transform does not, however, appear to have anything to do with schemata, deception, or representing the objective function (the interested reader is referred to chapter 2, contributed by Vose and Wright, in the book "Genetic Algorithms for Pattern Recognition", 1996).

The remainder of this talk is aimed at uncovering, in general terms, the inherent emergent behavior of random heuristic search. Towards that end, presentation of the specific details of $\mathcal{G}$ (which would specialize it to the simple genetic algorithm) will be postponed in favor of pursuing

The expression

$$\sum \left(\ln \sqrt{2\pi r q_j} + \frac{1}{12 r q_j + \theta(r q_j)} \right)$$

measures the *dispersion* of the population vector q. A minimally disperse population is homogeneous (r identical population members) and corresponds to $q = e_i$ for some i (where e_i is the ith column of the identity matrix). The corresponding dispersion is $O(\ln r)$. If $n \geq r$, a maximally disperse population has no duplication (q has r nonzero components which are all $1/r$) and dispersion r. The factor

$$\exp \left\{ - \sum \left(\ln \sqrt{2\pi r q_j} + \frac{1}{12 r q_j + \theta(r q_j)} \right) \right\}$$

occurring on the right hand side of theorem 10.2 indicates the probability that q is the next generation decays exponentially with increasing dispersion.

The combined effect of the two influences of discrepancy and dispersion is that random heuristic search is biased towards a less dispurse population near the expected next generation.

11. Markov Chain. The characterization of random heuristic search completed in the previous section was in terms of the sequence

$$p, \quad \tau(p), \quad \tau^2(p), \quad \ldots$$

produced by sampling Ω. This can be thought of as a sequence of random vectors (p may be regarded as the random vector which with probability one assumes only the single value p). Moreover, each random vector in the sequence depends only on the value of the preceding one. Such a sequence is called a *Markov chain.* The matrix defined by

$$Q_{p,q} \quad = \quad r! \prod \frac{(\mathcal{G}(p)_j)^{r q_j}}{(r q_j)!}$$

for $p, q \in \frac{1}{r} X_n^r$, is the *transition matrix* of the Markov chain. By theorem 10.2, $Q_{p,q}$ is the probability that $\tau(p) = q$. The definition of τ in the previous section rests ultimately on Ω as the induced map

$$\begin{tikzcd} p \arrow[r, dashed] \arrow[d] & \tau(p) \\ \mathcal{G}(p) \arrow[ur, "sample\ \Omega"'] & \end{tikzcd}$$

This conceptualization can now be replaced by an abstraction which makes no reference to Ω at all: from current configuration p, produce $q = \tau(p)$ with probability $Q_{p,q}$.

consequences suggested by the structural properties of the framework just presented.

13. Large populations. This section is mainly about the relationship between random heuristic search and its heuristic, as population size goes to infinity. A consequence of the results obtained is a view of a simple genetic algorithm's transient and asymptotic behavior in the large population case, given a well behaved heuristic.

Before proceeding, RHS algorithms are first classified according to the behavior of $\mathcal{G}$. An instance of random heuristic search is *focused* if $\mathcal{G}$ is continuously differentiable and for every $p \in \Lambda$ the sequence

$$p, \quad \mathcal{G}(p), \quad \mathcal{G}^2(p), \quad \ldots$$

converges. In this case $\mathcal{G}$ is also called focused. In terms of search, the latter condition means that the path determined by following at each generation what τ is expected to produce will lead to some state x. By the continuity of $\mathcal{G}$, such points x satisfy $\mathcal{G}(x) = x$ and are therefore *fixed points* of $\mathcal{G}$. In the case of the simple genetic algorithm, there are no known counter examples to $\mathcal{G}$ being focused when the mutation rate is less than 1/2. Moreover, Vose and Wright have proved that $\mathcal{G}$ is focused if proportional selection is used, the mutation rate is small, and the objective function has low epistasis.

An instance of random heuristic search is *hyperbolic* if $\mathcal{G}$ is continuously differentiable and its differential $d\mathcal{G}_x$ at x has no eigenvalues of unit modulus when x is a fixed point. In this case $\mathcal{G}$ is also called hyperbolic. In the case of the simple genetic algorithm, it has been proved by Eberlein and Vose that if proportional selection is used, then the set of fitness functions for which $\mathcal{G}$ is hyperbolic is dense and open (this was Mary Eberlein's Ph.D. dissertation). Generic hyperbolicity is believed to be the general case.

An instance of random heuristic search is *ergodic* if the Markov chain which represents it is ergodic for all $r > 0$. In this case $\mathcal{G}$ is also called ergodic. In the case of the simple genetic algorithm, ergodicity is insured by positive mutation.

The remainder of the talk deals with focused, hyperbolic, ergodic random heuristic search. It will turn out that fixed points are particularly relevant to both the transient (short term) and asymptotic (long term) behavior.

14. Transient and steady state behavior. The next theorem shows as $r \to \infty$ that, with probability converging to 1, the transient behavior of a population trajectory converges to the path determined by iterates of $\mathcal{G}$, and the initial transient occupies a time span diverging to infinity.

THEOREM 14.1. *Given $k > 0$, $\varepsilon > 0$ and $\gamma < 1$, there exists N such that with probability at least γ and for all $0 \leq t \leq k$*

$$r > N \implies \|\tau^t(x) - \mathcal{G}^t(x)\| < \varepsilon$$

Theorem 14.1 suggests (for large r) that some aspects of steady state behavior may be manifestations of transient behavior when $\mathcal{G}$ is focused. Let π be the probability measure corresponding to the steady state distribution of random heuristic search,

$$\pi(A) \;=\; \sum_{v \in \frac{1}{r} X_n^k} x_v \, [v \in A]$$

where x is the steady state distribution (probability vector) satisfying $x^T = x^T Q$ and [*expression*] denotes 1, if *expression* is true, and 0 otherwise. Thus $\pi(A)$ represents the proportion of time that populations spend in A, averaged over infinitely many generations. Since π is, for each population size r, a probability measure over the compact set Λ, a theorem of Prokhorov implies that every infinite sequence of π (corresponding to a sequence of r) has an infinite subsequence which converges weakly to some probability measure π'. Passing to the subsequence, this means that for every continuous function $h : \Lambda \rightarrow [0, 1]$,

$$\int h \, d\pi \;\longrightarrow\; \int h \, d\pi'$$

Let $\Im$ be the set of fixed points of $\mathcal{G}$. The next theorem provides a partial answer to how transient behavior influences steady state behavior.

THEOREM 14.2. *Suppose $\mathcal{G}$ is focused and ergodic. For every open set U containing $\Im$,*

$$\lim_{r \to \infty} \pi(U) \;=\; 1$$

In the large population case, theorem 14.2 indicates where population trajectories predominately spend time; near fixed points of $\mathcal{G}$. Moreover, theorem 14.1 indicates that a trajectory from x moves towards a fixed point of $\mathcal{G}$ by approximately following the path $x, \mathcal{G}(x), \mathcal{G}^2(x), \ldots$ The the next section investigates how quickly this path approaches a fixed point.

15. Logarithmic convergence. The definition of logarithmic time to convergence faces several obstacles. Perhaps the most obvious is the existence of a sequence of initial populations along which the time to convergence diverges to infinity. To circumvent this difficulty, let a probability density ρ be given over Λ and define the probability that the initial population is contained in A as

$$\int_A \rho \, d\lambda$$

where λ is Lebesgue measure. The task is then to show that for every ρ and every $\varepsilon > 0$, there exists a set A of probability at least $1 - \varepsilon$ such that if the initial population is in A, then the time to convergence is logarithmic.

The next difficulty is that, typically, a orbit under $\mathcal{G}$ will never reach the limit it is approaching. It is natural, therefore, to let $0 < \delta < 1$ denote how close trajectories are required to get to the limit, and then to require that they do so, within $O(-\ln \delta)$ generations.

In summary, *logarithmic convergence* is defined as follows: for every probability density ρ and every $\varepsilon > 0$, there exists a set A of probability at least $1 - \varepsilon$ such that for all $x \in A$ and $0 < \delta < 1$, the number k of generations required for $\|\mathcal{G}^k(x) - \omega(x)\| < \delta$ is $O(-\ln \delta)$, where $\omega(x)$ denotes the limit of $\mathcal{G}^t(x)$ as $t \to \infty$.

The next theorem makes use of the following technical condition. $\mathcal{G}$ is said to be *regular* if whenever C has measure zero, then so does the set $\mathcal{G}^{-1}(C)$. In the case of the simple genetic algorithm, Vose and Wright have proved that $\mathcal{G}$ is regular if the crossover rate is less than 1 and the mutation rate is strictly between 0 and 1/2 (provided that every string has positive selection probability).

THEOREM 15.1. *If $\mathcal{G}$ is focused, hyperbolic, and regular, then $\mathcal{G}$ is logarithmically convergent.*

16. Punctuated equilibria. Assuming $\mathcal{G}$ is focused, hyperbolic, ergodic, and regular, the view of RHS behavior that emerges for the large population case is the following.

As $r \to \infty$, and then with probability converging to 1, the initial transient of a population trajectory converges to following the path determined by iterates of $\mathcal{G}$, and that transient occupies a time span diverging to infinity. Consequently, populations will predominately appear near some fixed point ω of $\mathcal{G}$ since the path $x, \mathcal{G}(x), \mathcal{G}^2(x), \ldots$ approaches a fixed point relatively quickly.

This appears in contrast to the fact that the RHS is an ergodic Markov chain; every state must be visited infinitely often. This is reconciled in *punctuated equilibria*: Random events may eventually move the system to a population x' contained within the basin of attraction (with respect to the underlying dynamical system corresponding to $\mathcal{G}$) of a different fixed point ω'. Since the behavior of random heuristic search is time independent, the anticipated behavior follows the trajectory $x', \mathcal{G}(x'), \mathcal{G}^2(x'), \ldots$ – as if x' were the initial population – to reach a new temporary stasis in the vicinity of ω'.

This cycle of a period of relative stability followed by a sudden change to a new dynamic equilibrium is the picture provided by the results of this section. It is an open question as to how large r must be before these qualitative aspects of RHS are typically exhibited in the general case.

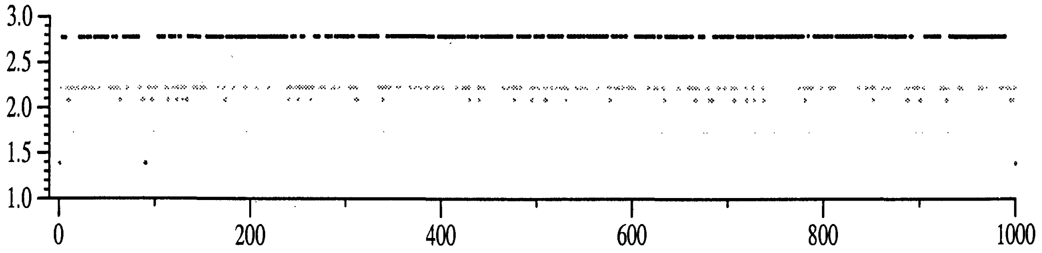
It is of interest that this phenomenon (punctuated equilibria) has been observed in practice when using GAs on optimization tasks. It is not uncommon for a GA to be described as undergoing a period of relative stability, after which it "discovers" a better solution which transforms the population. Neither is it uncommon for several such cycles to be manifest during long optimization runs. These observations are compatible with the conjecture that simple genetic search, as commonly used in practice, is influenced by the fixed points of the underlying dynamical system (corresponding to $\mathcal{G}$ on Λ).

17. Empirical evidence. This section considers a few computational examples for the simple genetic algorithm. Unlike the previous section whose results were based on arbitrarily large populations, the phenomena documented here were observed subject to a constraint on r. These examples took $r \approx \sqrt{n}$. Further empirical study is required (or better theorems need to be proved) to sort out the required linkage between search space size and population size in order for the behavior presented below to typically emerge. The conjecture that a logarithmic coupling (or some power of a log) might be appropriate for a wide class of objective functions is not incompatible with the empirical evidence.

When considering emergent behavior, perhaps the most fundamental question is: Where in Λ is the simple genetic algorithm at time t? Since the state space Λ has dimensionality too large for direct visualization (except in the case $\ell = 2$), alternate means of monitoring the progression from one generation to the next are required. A primitive means of reducing dimensionality is by measuring distance from populations to a reference point, say to the center $\mathbf{1}/n$ of Λ. The following graph shows what this looks like for string length 4, a random fitness function, (one-point) crossover rate 0.8, and mutation rate 0.01. Motivated by theoretical observations following theorem 10.2, the vertical axis measures distance as discrepancy,

$$distance(x, y) = \sum x_j \ln \frac{x_j}{y_j}$$

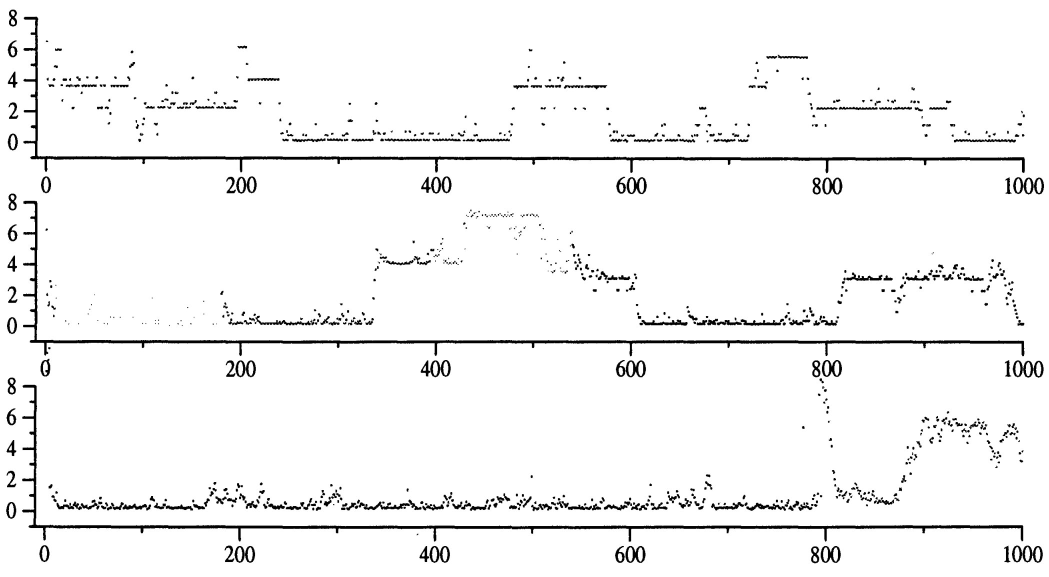
and the horizontal axis spans 1,000 generations. This example has population size 4 and corresponding state space of 3,876 populations in a simplex of dimension 15. The initial population p is random, and subsequent generations are produced by τ.



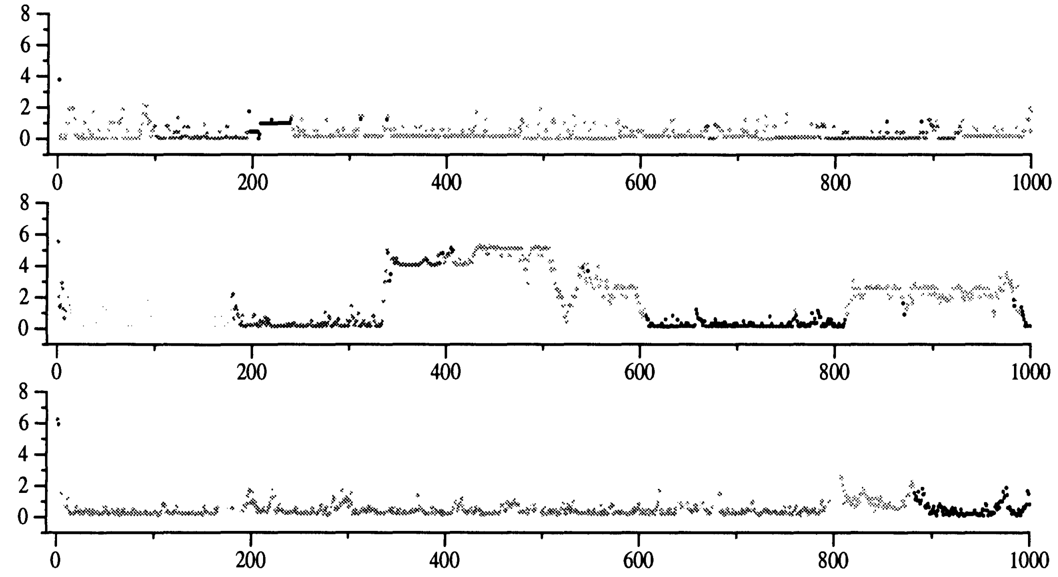
Certainly populations equidistant from $\mathbf{1}/n$ (i.e., with equal entropy) need not be near each other, but nevertheless there may be relatively few regions where the SGA is spending most of its time. One natural way to explore this is to locate regions where it seems reasonable that the SGA could be spending time, and then plot distance from the current population to such a place. The result should be essentially flat since at a typical generation (abscissa), the distance to one of these regions (ordinate) should be small.

Candidate regions are suggested by the discussion following theorem 10.2. Likely next generations are strongly related to the expected next generation. Near a fixed point, expected behavior is for the next generation to be near the current one, so fixed points of $\mathcal{G}$ may indicate areas where there is little pressure for change. It is plausible that the SGA could spend more time near such regions of Λ.

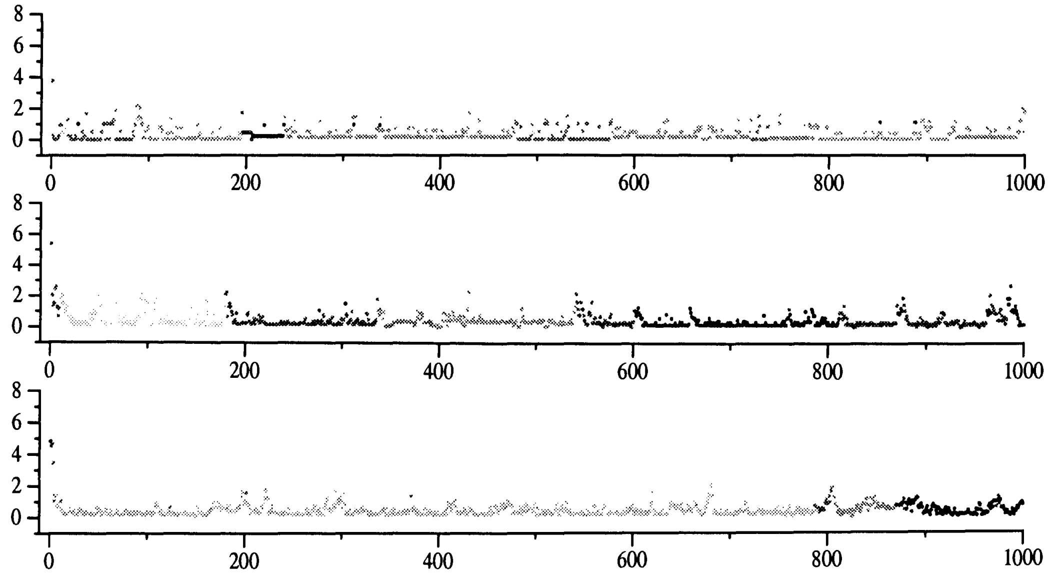
One method of locating attracting fixed points is by iterating $\mathcal{G}$. The following series of graphs shows generations (horizontal) versus distance to the stable fixed point to which iterates of $\mathcal{G}$ converge (vertical). The initial population p is random, and subsequent generations are produced by τ. The fitness function is random, a (one-point) crossover rate of 0.8 and mutation rate of 0.01 is used, and 1,000 generations are spanned. The string length ℓ begins at 4, increasing by 2 with each graph. The population size is approximately $\sqrt{n}$.



Given the complexity of these graphs, one might suspect fixed points of $\mathcal{G}$ do not help explain emergent behavior after all. However, the possibility remains that the SGA is particularly adept at seeking out fixed points of $\mathcal{G}$, more so than is the method of iterating $\mathcal{G}$ to locate them. The plateaus in these graphs suggest populations which could be concentrated in some localized region of Λ, perhaps a region near a fixed point not found by iterating $\mathcal{G}$. Extending the domain of $\mathcal{G}$ to the real affine space containing the simplex, using a minimization method to locate fixed points – whether stable or not – and then measuring distance to the nearest fixed point results in the following.



The majority of fixed points are unstable, outside the simplex, and near a vertex. The instability of population p within Λ near such a fixed point may be counterbalanced by the preference of RHS for populations having low dispersion. Another counterbalancing influence is the coarseness of the lattice $\frac{1}{r}X_n^r$ of points available to populations for occupation. Since $\mathcal{G}(p)$ is nearly p, the influence of discrepancy favors the next generation being identical to p, and this influence grows with increasing coarseness of the lattice. Even though overall the graphs are much lower, indicating a typical population's proximity to some fixed point, a few regions remain far from any fixed point. The next series of graphs is as the previous, except the fixed point equations were considered over *complex* space.

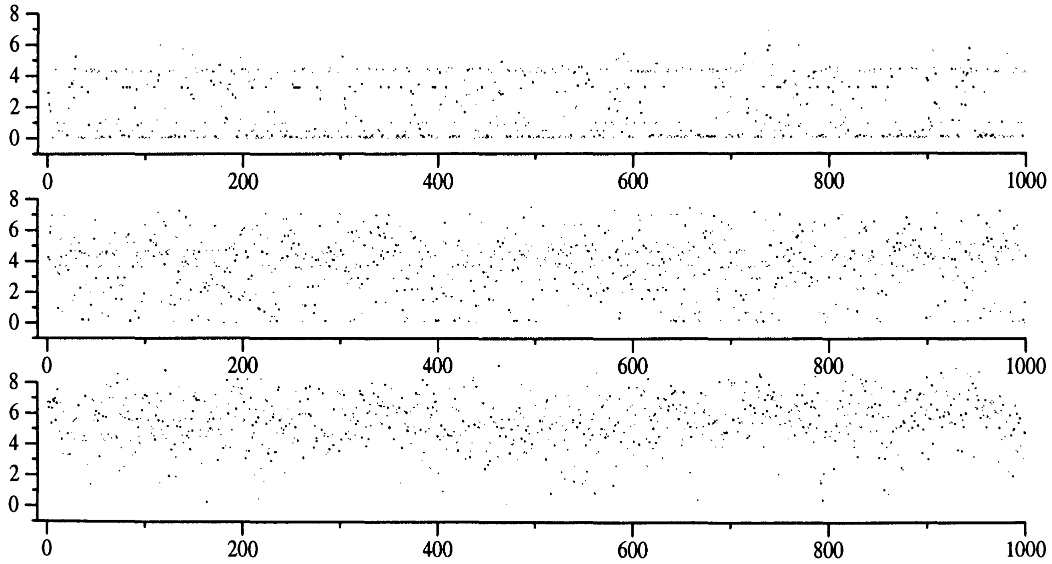


All graphs exhibit small height now that complex fixed points are included. Even though these examples are anecdotal, they are representative of several thousands of random cases.

It should not be supposed that irregularities in the graphs indicate an inability of fixed points to partially explain behavior. Finding fixed points is a difficult task exacerbated by high dimensionality which increases exponentially with ℓ. It is unlikely that all relevant fixed points have been found.

On the other hand, it is equally uncertain whether fixed points will continue to exert such a pronounced influence over emergent behavior as string length increases. Uncertainty revolves around the linkage between population size and string length. Whereas the large population case is fairly well understood (there the answer is yes), the small population case is not, and it is precisely that case which is the more interesting one from the point of view of search/optimization.

A natural question is whether the graphs presented in this section would retain their general appearance if, while leaving the fitness functions unchanged, alternate initial populations or different seeds for the random number generator were used. The answer is typically yes; sufficiently many fixed points corresponding to each fitness function were found so that populations are generally nearby. This begs the question of whether, with many fixed points available, a random point in Λ would be close to one of them. If that were the case, then these graphs have little meaning. The following sequence of graphs address this issue. Distance is measured to the nearest fixed point from random points having the same entropy distribution as populations in the previous graphs.



18. Summary. The evidence presented seems to indicate that simple genetic search is particularly adept at locating regions in the vicinity of fixed points of $\mathcal{G}$. This picture is conjectural, however, since no theorems were proved. Although smaller population sizes, $r = O(\ln n)$ for example, may seem more realistic, there is insignificant difference between $3\log_2 n$ and $\sqrt{n}$ for the string lengths considered $(3 < \ell < 9)$.

While different in important respects, the results presented here are reminiscent of the large population case. The increased influence of the dispersion term (in theorem 10.2) on transition probabilities, given smaller r, may contribute to increased importance of unstable and complex fixed points located near vertices of Λ. Because $\mathcal{G}$ is continuous, such fixed points locate regions where there is decreased pressure for change. The natural preference of RHS for low dispersion may counterbalance the instability of the underlying dynamical system (corresponding to $\mathcal{G}$) for populations in such regions. Moreover, the coarseness of the lattice of points available for occupation also serves to counterbalance the instability.

19. Small populations: Signal and noise. The previous considerations may seem, from a practitioner's viewpoint, academic. There are several reasons for this. One is that it is the initial transient (or first few) of the small population case that is of primary interest. Another is that s/he has a problem to solve, and wants an answer: Should a GA be used? If genetic search *is* being used, then *how* can it be made effective?

In general, I believe it can be shown that categorical answers to questions like these are of comparable hardness to finding the optimum by enumeration (again, there is no "free lunch"). The best one can hope for is results concerning a specific class of functions, and even then, the problem of deciding whether a general function belongs to the class is a hard problem. Nevertheless, it is quite likely that simple qualitative results providing insight into the general mechanism of RHS for all population sizes are near at hand.

Consider, for example, the following asymptotic result (as $r \to \infty$). Let $q = \mathcal{G}(p)$ and let C be an n by $n-1$ matrix having orthonormal columns perpendicular to $h = <\sqrt{q_0}, \ldots, \sqrt{q_{n-1}}>$.

THEOREM 19.1. *If $\mathcal{G}$ maps into the interior of Λ, then for any open subset U of $\mathbf{1}^\perp$,*

$$Pr\{\tau(p) \in \mathcal{G}(p) + U/\sqrt{r}\} = (2\pi)^{-(n-1)/2} \int_{C^T \operatorname{diag}(h)^{-1} U} e^{-y^T y/2}\, dy \; + \; o(1)$$

If an error term was provided for this result (i.e., an explicit form for "$o(1)$" in terms of r and n), a fairly simple decomposition of τ into a deterministic *signal* component, given by $\mathcal{G}$, and a stochastic *noise* component, given by the multinormal distribution, would result.

If one does not mind the multinomial distribution, theorem 10.2 shows, for any r, that $\tau(p)$ is given by a single sample from a multinomial distribution having mean $\mathcal{G}(p)$. Again, $\mathcal{G}(p)$ emerges as the deterministic *signal.* The stochastic *noise* component is given by the multinomial distribution.

Note how, in this decomposition into signal and noise, the signal is *invariant* in the sense that it is *independent* of population size. The noise is partially characterized by the following theorem.

THEOREM 19.1. *Let $\mathcal{E}$ denote expectation.*

$$\mathcal{E}(\| \tau(p) - \mathcal{G}(p) \|^2) = (1 - \| \mathcal{G}(p) \|^2)/r.$$

What complicates the small population case is threefold. First, as r decreases, the noise component increases. Second, the relative influence of dispersion grows. Third, the lattice of allowable values for population vectors, $\frac{1}{r} X_n^r$, becomes increasingly coarse, as fewer points, located in lower dimensional faces of Λ, become available for occupation. Genetic search is conducted in a low dimensional "skeleton" of Λ which constrains the system's ability to follow the signal.

It is of interest to understand which directions, or pathways through this skeleton, are more probable than others. In particular, one would like to know what strings are typically encountered while traversing the pathways. At this point, there are no proven simple answers (there is, of course, the Markov chain, but that quickly becomes unwieldy as r increases).

20. Metalevel chain. Assuming simple genetic algorithms are adept at locating regions in the vicinity of fixed points of $\mathcal{G}$, the transition probabilities from one such region to another are significant. In that case, simple genetic search could be modeled by a Markov chain over the fixed points. If the transition probabilities from temporary stasis in the vicinity of one fixed point to temporary stasis near another can be determined, then some aspects of the punctuated equilibria could in principle be analyzed.

The goal of constructing a meta level Markov chain, as described in the previous paragraph, has been achieved for general random heuristic search (subject to the technical conditions that $\mathcal{G}$ is hyperbolic, ergodic, and has a complete Lyapunov function), but only in the large population case.

Let $\rho = x_0, \ldots, x_k$ be a sequence of points from Λ, referred to as a *path* of length k from x_0 to x_k. The *cost* of ρ is

$$| \rho | = \alpha_{x_0, x_1} + \cdots + \alpha_{x_{k-1}, x_k}$$

where

$$\alpha_{u,v} = \sum v_j \ln \frac{v_j}{\mathcal{G}(u)_j}$$

and it is assumed that $\mathcal{G}$ maps Λ into its interior so as to avoid division by zero. Let the stable fixed points of $\mathcal{G}$ in Λ be $\{\omega_0, \ldots, \omega_w\}$ and define

$$\rho_{\omega_i, \omega_j} = \inf \{ | \rho | : \rho \text{ is a path from } \omega_i \text{ to } \omega_j \}$$

Let $\mathcal{C}_r$ be a Markov chain defined over $\{1, \ldots, w\}$ with $i \to j$ transition probability (for $i \neq j$)

$$\exp\{-r \, \rho_{\omega_i, \omega_j} + o(r)\}$$

Up to the uncertainly in the $o(r)$ terms, the desired Markov chain is C_r.

As section 4 demonstrates, C_r cannot possibly be appropriate for small r, because unstable, complex, and stable fixed points outside Λ make no contribution to C_r. Nevertheless, the form of the transition probabilities above is instructive. The likelihood of a transition from i to j is determined by the minimal cost path from ω_i to ω_j where a path incurs cost to the extent that it is made up of steps which end at a place differing from where $\mathcal{G}$ maps their beginning. By theorem 10.2, this roughly corresponds to the probability of the most likely sample path leading from ω_i to ω_j in the Markov chain with transition matrix Q. In other words, when r is large, and aside from the dispersion being unimportant, behavior is determined by a *single* sample path, rather than a *sum* over sample paths.

Is not unthinkable that in order to make significant progress in the small population case, the *particular* nature of $\mathcal{G}$ will have to be brought into play; perhaps general properties (hyperbolicity, ergodicity, etc.) will not suffice. Either way, this points to the pivotal importance of $\mathcal{G}$ – either in terms of its general properties, or else in terms of the details of its specific nature – and argues for its study as an abstract mathematical object.

21. The SGA's heuristic. This section presents a special case of the simple genetic algorithm's heuristic. It is not feasible to summarize a nontrivial part of all that is currently known, so simple *definition* of a particular case will have to do for this talk. For simplicity, the binary case will be described (Koehler, Bhattacharyya, and Vose have extended results to the general cardinality case) and only proportional selection will be considered.

For positive integer ℓ, the set of length ℓ binary strings is the Cartesian product

$$\Omega = \underbrace{Z_2 \times \cdots \times Z_2}_{\ell \text{ times}}$$

Since the ℓ-digit binary representations of integers in the interval $[0, 2^\ell)$ coincide with the elements of Ω, they are regarded as being the same. Elements of Z_2 form a finite field under the operations of addition and multiplication modulo 2

$\oplus$	0	1
0	0	1
1	1	0

$\otimes$	0	1
0	0	0
1	0	1

These operations are extended to Ω by applying them coordinate-wise. By convention, $\otimes$ takes precedence over $\oplus$, and both bind more tightly than operations which are not modulo 2.

For $x \in \Omega$, let $\overline{x}$ abbreviate $\mathbf{1} \oplus x$. In standard computer science nomenclature, $\oplus$ is *exclusive-or* on integers, $\otimes$ is *and*, and $x \mapsto \overline{x}$ is *not*. Note that $\otimes$ distributes over $\oplus$.

Let σ_k be the permutation matrix defined by

$$(\sigma_k)_{i,j} \quad = \quad [i \oplus j = k]$$

The permutation σ_k corresponds to applying the map $i \mapsto i \oplus k$ to subscripts. That is,

$$\sigma_k < x_0, \ldots, x_{n-1} > \quad = \quad < x_{0 \oplus k}, \ldots, x_{(n-1) \oplus k} >$$

22. Selection. The symbol s will be used for three equivalent (though different) things. This overloading of s does not take long to get used to because context makes meaning clear. The benefits are clean and elegant presentation and the ability to use a common symbol for ideas whose differences are often conveniently blurred.

First, $s \in \Lambda$ can be regarded as a *selection distribution* describing the probability s_i with which i is selected (with replacement) from the current population for participation in forming the next generation. A selected element is an intermediate step towards producing the next population, not typically a member of it. In total, $2r$ such selections will be made, the aggregate of which is sometimes referred to as the *gene pool.*

Second, $s : \Lambda \rightarrow \Omega$ can be regarded as a *selection function* which is nondeterministic. The result $s(p)$ of applying s to p is i with probability given by the ith component s_i of the selection distribution. Of course, for there to be a nontrivial dependence on p, the selection distribution must be some function $\mathcal{F}$ of p. The function $\mathcal{F}$ is referred to as the *selection scheme.*

Third, $s \in \Lambda$ can be regarded as a population vector.

In analogy with survival of the fittest, an integral part of $\mathcal{F}$ is a *fitness function* $f : \Omega \rightarrow \Re$ which can be used (in a variety of ways) to determine a selection scheme. The fitness function is assumed to be injective. The value $f(i)$ is called the *fitness* of i. Through the identification $f_i = f(i)$, the fitness function may be regarded as a vector.

Proportional selection refers to the selection function corresponding to the selection scheme

$$\mathcal{F}(x) \quad = \quad f \cdot x / f^T x$$

(where $f \cdot x$ denotes $\text{diag}(f)x$). When proportional selection is being used, it is assumed that the fitness function is positive. Since proportional selection is homogeneous, without loss of generality $f \in \Lambda$.

By letting the heuristic $\mathcal{G}$ be the selection scheme, results from previous sections apply to selection. For example, with population size $2r$, $\tau(p)$ becomes the gene pool. Invoking theorem 10.1, the expected gene pool is described by the population vector $s = \mathcal{F}(p)$. By definition, the selection distribution is $s = \mathcal{F}(p)$. Hence, as elements of Λ, the selection distribution is identical to the expected gene pool.

23. Mutation. The symbol μ will also be used for three different (though related) things.

First, $\mu \in \Lambda$ can be regarded as a distribution describing the probability μ_i with which i is selected to be a *mutation mask* (additional details will follow).

Second, $\mu : \Omega \to \Omega$ can be regarded as a *mutation function* which is nondeterministic. The result $\mu(x)$ of applying μ to x is $x \oplus i$ with probability given by the ith component μ_i of the distribution μ. The i occurring in $x \oplus i$ is referred to as a mutation mask. The application of μ to x is referred to as *mutating* x.

Third, $\mu \in [0, 0.5)$ can be regarded as a *mutation rate* which implicitly specifies the distribution μ according to the rule

$$\mu_i \;=\; (\mu)^{\mathbf{1}^T i}\,(1-\mu)^{\ell - \mathbf{1}^T i}$$

The distribution μ need not correspond to any mutation rate, although that is certainly the classical situation. Any element $\mu \in \Lambda$ whatsoever is allowed.

The effect of mutating x using mutation mask i is to alter the bits of x in those positions where the mutation mask i is 1. When mutation is affected by a rate, the probability of selecting mask i depends only on the number of 1s that i contains.

If the mutation rate is nonzero (the typical case), then every element of Ω has a positive probability of being the result of $\mu(x)$. Mutation is said to be *zero* if $\mu_i = [i = 0]$. For arbitrary $\mu \in \Lambda$, mutation is called *positive* if $\mu_i > 0$ for all i.

24. Crossover. It is convenient to use the concept of *partial probability* . Let $\zeta : A \to B$ and suppose that $\phi : A \to [0,1]$. To say "$\xi = \zeta(a)$ with partial probability $\phi(a)$" means that $\xi = b$ with probability $\sum_a [\zeta(a) = b]\, \phi(a)$.

The description of crossover parallels the description of mutation given in the previous section; the symbol χ will be used for three different (though related) things.

First, $\chi \in \Lambda$ can be regarded as a distribution describing the probability χ_i with which i is selected to be a *crossover mask* (additional details will follow).

Second, $\chi : \Omega \times \Omega \to \Omega$ can be regarded as a *crossover function* which is nondeterministic. The result $\chi(x, y)$ is $x \otimes i \oplus \bar{i} \otimes y$ with partial probability $\chi_i/2$ and is $y \otimes i \oplus \bar{i} \otimes x$ with partial probability $\chi_i/2$. The i occurring in the definition of $\chi(x, y)$ is referred to as a crossover mask. The application of $\chi(x, y)$ to x, y is referred to as *recombining* x and y.

The arguments x and y of the crossover function are called *parents*, the pair $x \otimes i \oplus \bar{i} \otimes y$ and $y \otimes i \oplus \bar{i} \otimes x$ are referred to as their *children*. Note that crossover produces children by exchanging the bits of parents in those positions where the crossover mask i is 1. The result $\chi(x, y)$ is called their *child.*

Third, $\chi \in [0, 1]$ can be regarded as a *crossover rate* which specifies the distribution χ according to the rule

$$\chi_i = \begin{cases} \chi c_i & \text{if } i > 0 \\ 1 - \chi + \chi c_0 & \text{if } i = 0 \end{cases}$$

where the distribution $c \in \Lambda$ is referred to as the *crossover type.* Classical crossover types include *1-point crossover*, for which

$$c_i = \begin{cases} 1/(\ell - 1) & \text{if } \exists k \in (0, \ell) \,.\, i = 2^k - 1 \\ 0 & \text{otherwise} \end{cases}$$

and *uniform crossover*, for which $c_i = 2^{-\ell}$. However, any element $c \in \Lambda$ whatsoever is allowed.

Obtaining child z from parents x and y via the process of mutation and crossover is called *mixing* and has probability denoted by $m_{x,y}(z)$.

THEOREM 24.1. *If mutation is performed before crossover, then*

$$m_{x,y}(z) = \sum_{i,j,k} \mu_i \mu_j \frac{\chi_k + \chi_{\overline{k}}}{2} [(x \oplus i) \otimes k \oplus \overline{k} \otimes (y \oplus j) = z]$$

If mutation is performed after crossover, then

$$m_{x,y}(z) = \sum_{j,k} \mu_j \frac{\chi_k + \chi_{\overline{k}}}{2} [x \otimes k \oplus \overline{k} \otimes y = z \oplus j]$$

The *mixing scheme* $\mathcal{M} : \Lambda \to \Lambda$ is defined by the component equations

$$\mathcal{M}(x)_i = x^T \sigma_i M \sigma_i x$$

THEOREM 24.2. *The heuristic $\mathcal{M}$ corresponds to the instance of RHS which produces elements for the next generation by mixing the results of uniform choice (with replacement) from the population.*

25. SGA's Heuristic. The simple genetic algorithm's heuristic is the composition of mixing and selection

$$\mathcal{G} = \mathcal{M} \circ \mathcal{F}$$

which depends on the three control parameters, $f, \mu, \chi \in \Lambda$. This completes the definition of the simple genetic algorithm as an instance of random heuristic search.

26. Closing remarks. What is known about the theory of the SGA's heuristic has only been hinted at in this talk. The forthcoming book "The Simple Genetic Algorithm: Foundations and Theory" (Mit Press, in press) will be a good place for the interested reader to begin.

I would like to indicate where the example of section 1.3.1 (Static Schema Analysis) came from. The key is that, in the zero mutation case, the spectrum of the differential $d\mathcal{G}_x$ is known explicitly at every fixed point x which corresponds to an absorbing state of the Markov chain. For further details, see "Stability of Vertex Fixed Points and Applications" (Vose and Wright) in the book "Foundations of Genetic Algorithms III" (Whitley & Vose, Editors).

I owe a lot to friends, colleagues, and the National Science Foundation, for supporting this line of inquiry which so radically departs from the classical schemata-based "GA theory." Many have provided inspiration and companionship along the way, but Alden Wright and Gary Koehler deserve special mention.