International Journal of Pure and Applied Mathematics

Volume 61 No. 3 2010, 241-251

A SIMPLE MODEL FOR GENETIC ALGORITHM CONVERGENCE

Martin Dowd

1613, Wintergreen Pl.
Costa Mesa, CA 92626, USA
e-mail: MartDowd@aol.com

Abstract: A simple model of genetic algorithm convergence is presented. An "ideal" iteration may be specified, in terms of a random variable for the weight distribution. The behavior of an "actual" iteration may in some cases be sufficiently close to the ideal behavior that useful quantitative information is obtained. In any case, a useful framework for analyzing a genetic algoritm is provided.

AMS Subject Classification: 68W40
Key Words: genetic algorithm convergence, iteration, "ideal" iteration, "actual" iteration

1. Introduction

Genetic algorithms are acknowledged to be a useful optimization method; [6] contains a list of over 50 application areas. Various theoretical frameworks have been devised for analyzing their behavior (see [2], [3]), [7] has a discussion of two well-known frameworks, schema and Markov chains. Here a new framework will be presented. Related results may be found in [5]. Although this framework is appearing here for the first time, it was originally considered by the author in 1995.

A discussion of GAs (genetic algorithms) will be omitted here, since many such are readily available; rather the specific algorithms under consideration will be presented without discussion. The following will be assumed.

— The alphabet is $\{0, 1\}$.

Received: January 28, 2010

— The string length is l.

— The population size is s.

— The weight, or fitness, function (denoted w) is assumed to take values from a finite range of nonnegative integers.

— The proportional selection and mutation operations are considered.

2. Ideal Proportional Selection

In an ideal GA, the weight is a random variable, which evolves with time. Let W_t denote it at generation t, where $t = 0$ for the initial generation.

Consider first a GA with proportional selection only. In an actual GA, for a given string x, the number of copies of x at time $t + 1$ equals the number at time t, times $(w(x)/W)s$ where $W = \sum_x w(x)$. Also, W is approximately s times the mean weight at generation t. Letting $P_t(w)$ denote $P\{W_t = w\}$, the recursion

$$P_{t+1}(w) = wP_t(w)/\sum_w wP_t(w) \tag{1}$$

may be assumed to hold.

Theorem 1.

$$P_t(w) = w^t P_0(w) \;/\; \sum_w w^t P_0(w). \tag{2}$$

Proof. This follows easily from (1), by induction on t.

P_t depends only on P_0, and does not depend on any other parameters. Clearly this is an "ideal" behavior, and "actual" GA searches only approximate it.

Let E_t^k denote the k-th moment $E(W_t^k)$ of W_t. Using (2), it follows by algebra that

$$E_t^k = E_0^{k+t}/E_0^k. \tag{3}$$

□

3. The Binomial Distribution

In this section $P_0(w)$ will be taken to be the binomial distribution

$$P_0(w) = 2^{-l}\binom{l}{w},$$

where l and w are integers with $l > 0$ and $0 \le w \le l$. This is an important example; it corresponds to a GA on bit strings where the weight is the number of bits in agreement with a single "target", with random initial strings.

The distribution P_t depends also on the parameter l, and when necessary will be denoted P_{tl}, with similar notation for other values. Using (2) it is easily seen that

$$P_t(w) = w^t\binom{l}{w} \Big/ S_t, \quad \text{where} \quad S_t = \sum_w w^t\binom{l}{w}.$$

Also,

$$E_0^t = S_t/2^t. \tag{4}$$

This distribution might be called a "power binomial" distribution.

The following additional notation is introduced. Let $\mathbf{w}$ denote w/l, $\mathbf{t}$ denote t/l, etc. In some contexts $\mathbf{t}$ or $\mathbf{w}$ must be rational, and l a multiple of the denominator; such will be assumed without comment when necessary. $\bar{w}_t$ will be used synonymously with E_t^1. Given $\mathbf{t}$, let $\boldsymbol{\mu}$ be the value such that $\mathbf{t} = \boldsymbol{\mu}(\log(\boldsymbol{\mu}) - \log(1-\boldsymbol{\mu}))$.

Theorem 2. *a. $P_t(w)$ is unimodal. In fact, there is a unique real number w with $0 \le w \le l$, which will be denoted $w_t^=$, such that*

$$\left(\frac{w+1}{w}\right)^t = \frac{w+1}{l-w}. \tag{5}$$

The value of w (or the smallest of two consecutive such) at which $P_t(w)$ is maximum satisfies $w_t^= \le w \le w_t^= + 1$. For $l \ge 1$ the value $w_t^=$ is strictly increasing with t.

b. For fixed $\mathbf{t}$, as $l \to \infty$, $\mathbf{w}^= \to \boldsymbol{\mu}$.

c. For $l \ge 2$ the mean $\bar{w}_t$ is strictly increasing with t.

d. Let $\sigma = \varsigma\sqrt{l}$, where

$$\varsigma = \sqrt{\frac{\boldsymbol{\mu}(1-\boldsymbol{\mu})}{1 + ((1-\boldsymbol{\mu})/\boldsymbol{\mu})\mathbf{t}}}.$$

Then for fixed $\mathbf{t}$ *and* $|x|$ *bounded by a constant,*

$$\begin{aligned}\log(P_{tl}(\mu + x\sigma)) = \quad & -\frac{1}{2}\log 2\pi - \frac{1}{2}(\log l + \log \boldsymbol{\mu} + \log(1-\boldsymbol{\mu})) \\ & +(\mathbf{t}\log\boldsymbol{\mu} - \boldsymbol{\mu}\log\boldsymbol{\mu} - (1-\boldsymbol{\mu})\log(1-\boldsymbol{\mu}))l \\ & +\mathbf{t}l\log l - \frac{x^2}{2} + O\left(\frac{1}{\sqrt{l}}\right).\end{aligned}$$

Proof. For part a, for $w < l$ $P_t(w+1)/P_t(w)$, equals the ratio of the left side of (5) to the right side. The left side is constantly 1 if $t = 0$, else decreases from ∞; further it is increasing with t for all w. The right side increases from $1/l$ to ∞. For part b, take the log of both sides of (5), expand $\log(1 + 1/w)$ in a Taylor series, and use $(l-1)/2 \le w^= < l$. For part c, using (3) and (4), $\bar{w}_t$ equals S_{t+1}/S_t, whence $\bar{w}1_{t+1} - \bar{w}1_t = (S_{t+2}S_t - S_{t+1}^2)/(S_tS_{t+1})$. Finally

$$S_{t+2}S_t - S_{t+1}^2 = \sum_{1\le w_1<w_2\le l} (w_2 - w_1)^2 \binom{l}{w_1}\binom{l}{w_2}.$$

Part d is proved by adapting the proof of the normal approximation to the binomial distribution, as found in [8] for example. □

In light of Theorem 2.d the following seems certain, but no further discussion will be given here.

Conjecture 3. *For fixed* $\mathbf{t}$*, as* $l \to \infty$*,* $\bar{\mathbf{w}} \to \boldsymbol{\mu}$*.*

For large $\mathbf{t}$, $\boldsymbol{\mu}$ is approximately $1 - e^{-\mathbf{t}}$ (the error is less than 1% for $\mathbf{t} \ge 3$). Figure 1 shows a graph of $\boldsymbol{\mu}$ versus $\mathbf{t}$. It also shows $\bar{\mathbf{w}}$ when $l = 16$, as dashed lines.

Observations suggest that the following is true.

Conjecture 4. *Let* $\Delta = \bar{w}_{tl} - w^=_{lt}$*. Then* $0 \le \Delta \le .5$*. For fixed* l Δ *strictly decreases to 0 for* $t \ge 1$*. For fixed* t Δ *strictly increases to .5 for* $l \ge 0$*.*

That $0 \le \Delta$ has been verified for $l \le 99$ and $t \le 5l$ (by verifying $(S_{t+1} + S_t)^{t-1}(l - S_{t+1}) \le S_{t+1}^t$ for $t > 0$). A check using double precision finds no exceptions to the entire conjecture, for these values. The program used for these checks (and other programs written during the writing of this paper) is available at the author's website, "www.hyperonsoft.com".

Let $\mathcal{S}_i(t,j)$ denote the Stirling numbers of the i-th kind, for $i = 1, 2$, as defined in [4].

Theorem 5. *a. For* $l, t > 0$*,* $S_{tl} = l(S_{t-1,l} - S_{t-1,l-1})$*.*

b. $S_t = 2^{l-t}\sigma_t$ *where* $\sigma_0 = 1$ *and for* $t > 0$*,* $\sigma_t(l) = l(2\sigma_{t-1}(l) - \sigma_{t-1}(l-1))$*.* σ_t *is a monic integer polynomial in* l*, of degree* t*.*

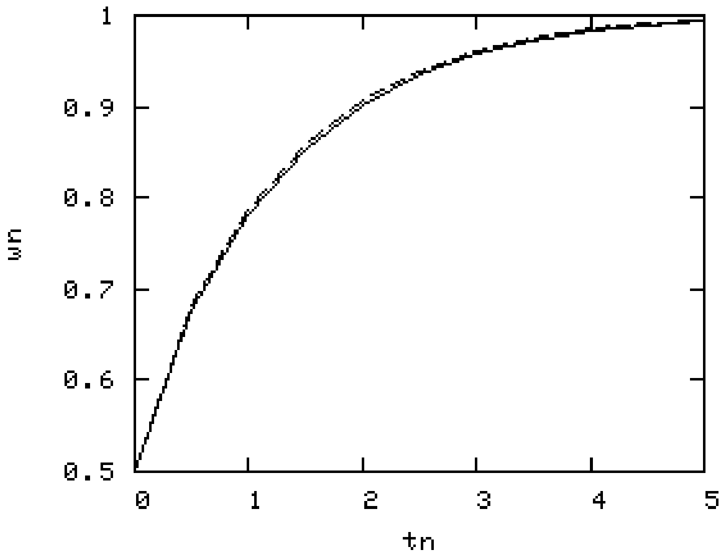

Figure 1: Behavior of normalized mode/mean

c.

$$\sigma_t = \sum_{k=0}^{t}\left(\sum_{j=k}^{t}(-1)^{j-k}2^{t-j}\mathcal{S}_2(t,j)\mathcal{S}_1(j,k)\right)x^k.$$

d. The coefficient of l^{t-1} in σ_t equals $\binom{t}{2}$.

Proof. Parts a and b appear in [1]. Part a follows using the identity $\binom{l}{w} = \frac{l}{w}\binom{l-1}{w-1}$ and the addition formula for binomial coefficients. Part b follows by induction on t. For part c, the j-th factorial moment $\mu_{(j)}$ of the binomial distribution equals $l^{(j)}/2^j$, where $l^{(j)}$ denotes the falling factorial. $l^{(j)}$ equals $\sum_{k=0}^{j}(-1)^{j-k}\mathcal{S}_1(j,k)x^k$. The t-th moment μ'_t about the origin equals $\sum_{j=0}^{t}\mathcal{S}_2(t,j)\mu_{(j)}$. $S_t = 2^l\mu'_t$, so by part b $\sigma_t = 2^t\mu'_t$, part d follows using part c. □

Theorem 6. *For fixed t, as $l \to \infty$, the following hold.*

a. $\bar{w} = (l+t)/2 + O(1/l)$.

b. $w^{=} = (l+(t-1))/2 + O(1/l)$.

Proof. Part a follows using $\bar{w}_t = \sigma_{t+1}/(2\sigma_t)$, Theorem 5.d, and polynomial division. Part b follows by standard asymptotic formulas, noting that $w \geq$

$l/2$. □

4. Other Distributions

In a concentrated distribution, $P_0(w) = 1$ if $w = c$, and $P_0(w) = 0$ otherwise. This distribution corresponds to a GA where the cost function is constantly c. P_t equals P_0 for all t.

In a uniform distribution, $P_0(w) = 1/(l+1)$ for $w = a + b(j/l)$ where $0 \le j \le l$. This arises from a GA where the weight increases linearly with the Euclidean distance, for example. Letting $S_t = (1/(l+1)) \sum (a + b(j/l))^t$ and using the approximation $S_t \approx \int_0^1 (a+bx)^t dx$,

$$\bar{w}_t \approx (a+b)\frac{1-\alpha^{t+2}}{1-\alpha^{t+1}}\frac{t+1}{t+2}, \quad \text{where} \quad \alpha = \frac{1}{a+b}.$$

Further discussion is omitted.

For the remainder of the paper, only the binomial distribution will be considered.

5. Mutation

The mutation operation is useful in actual GAs to “enrich the population”. It can be analyzed in an ideal GA. Closed forms are less easy to obtain, but computer programs can be written to obtain data.

To add the mutation operation to the idealized GA, let V_{t+1} denote the random variable giving the weight distribution during generation $t+1$, after the mutation operation but before the selection operation. Then

$$\begin{aligned} P\{V_{t+1} = v\} &= \sum_w M(w,v) P\{W_t = w\}, \\ P\{W_{t+1} = w\} &= \frac{wP\{V_{t+1} = w\}}{\sum_w wP\{V_{t+1} = w\}}, \end{aligned}$$

where $M(w,v)$ is the probability that a weight w string mutates to a weight v string.

Suppose a bit of a string is flipped with probability p by the mutation operation. For large l the probability that k bits of a string are flipped is approximately $e^{-\lambda}\lambda^k/k!$ where $\lambda = pl$. For small λ this in turn can be approximated by $1-\lambda$ for $k = 0$; λ for $k = 1$; and 0 for $k > 1$. For binary strings,

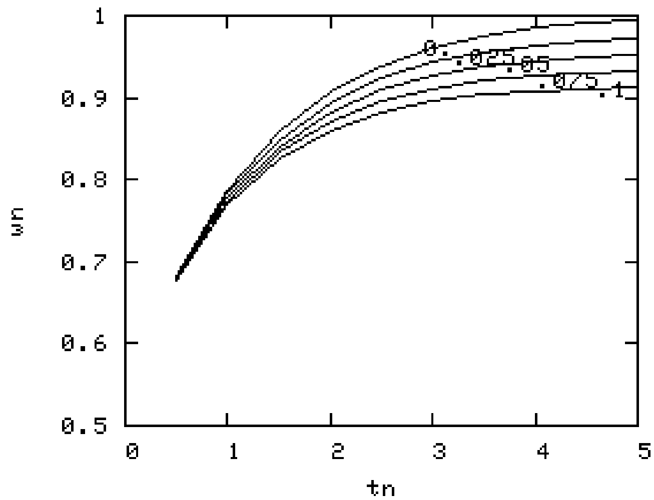

Figure 2: Dependence on mutation parameter

$M(w,v)$ can be taken as $1-\lambda$ if $v=w$; $\lambda(w/l)$ if $v=w-1$; $\lambda(1-w/l)$ if $v=w+1$; and 0 otherwise. λ must be sufficiently small; only values $\lambda \leq .1$ will be considered.

Using the above approximation, straightforward computation yields

$$E(V_{t+1}) = \left(1-\frac{2\lambda}{l}\right)E(W_t)+\lambda,$$
$$E(V_{t+1}^2) = \left(1-\frac{4\lambda}{l}\right)E(W_t^2)+2\lambda E(W_t)+\lambda.$$

$E(W_{t+1})$ equals $E(V_{t+1}^2)/E(V_{t+1})$.

No attempt will be made to obtain further formulas. It is worth mentioning, though, that if $\vec{P}_t$ denotes the vector of values $P_t(w)$, there are matrices M and S such that $\vec{P}_{t+1}=SM\vec{P}_t$, whence $\vec{P}_t=(SM)^t\vec{P}_0$.

Figure 2 shows how the mutation operation affects the behavior of the normalized mean $E(W_t)/l$ as a function of t/l, for $l=16$. Tests show that for larger values of l the dependence on l is slight.

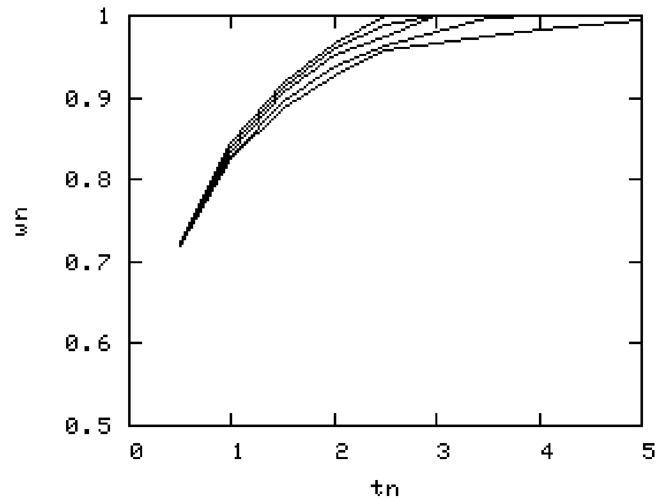

Figure 3: Dependence with $\kappa = .1$

6. Scaling

It is well-recognized that scaling the fitness function can improve the performance of a genetic algoritm. Ideal GAs provide a framework for evaluating scaling methods. A simple method discards a fraction κ of the lowest weight strings, after mutation and before selection.

In the ideal GA, values of w for which the cumulative distribution is less than κ are set to 0, and κ is subtracted from the first value for which this is possible. The distribution is then normalized by dividing by $1 - \kappa$.

Figures 3 and 4 duplicate Figure 2 (for which $\kappa = 0$) for the values $\kappa = .1$ and $\kappa = .2$ respectively. As can be seen, in the ideal GA at any rate this method yields substantial improvement.

7. Comparison with an Actual GA

An actual GA where the weight function is the Hamming weight is a useful case for theoretical studies. It is modeled by the ideal GA with an initial binomial distribution. It occurs in applications, and yields data relevant to other GAs.

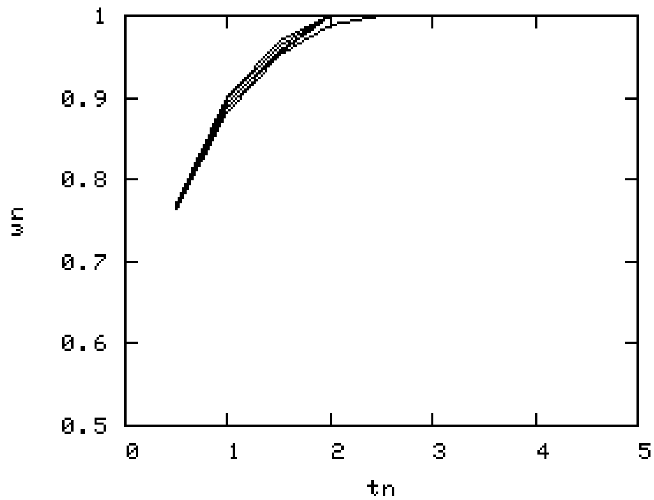

Figure 4: Dependence with $\kappa = .2$

An actual GA has an additional parameter, the population size s. Larger values should yield improved chances of finding high weight strings, while smaller values yield greater speed.

Other considerations include the following:

— With proportional selection only, only the maximum weight of the initial population can be achieved. Further, a population uniformly this weight might not be achieved, because there are other states which are fixed points.

— In the selection operation of an actual GA the replacement of x_i with $(w_i/W)S$ copies can only be done approximately. A convenient method is as follows. Conceptually replace x_i by w_iS copies, and take every W-th entry (starting, say, at the $\lfloor W/2 \rfloor$-th).

— The mutation operator adds the new strings to the population, which ensures that the highest weight strings do not get replaced. In the ideal GA, the actual mutation parameter λ should be replaced by $\lambda' = \lambda/(1-\lambda)$.

— In simulating a run of the actual GA, only the string weights need be maintained.

Results of various tests included the following.

— With $\lambda = .025$ and $\kappa = 0$, the actual GA gets stuck at the maximum of

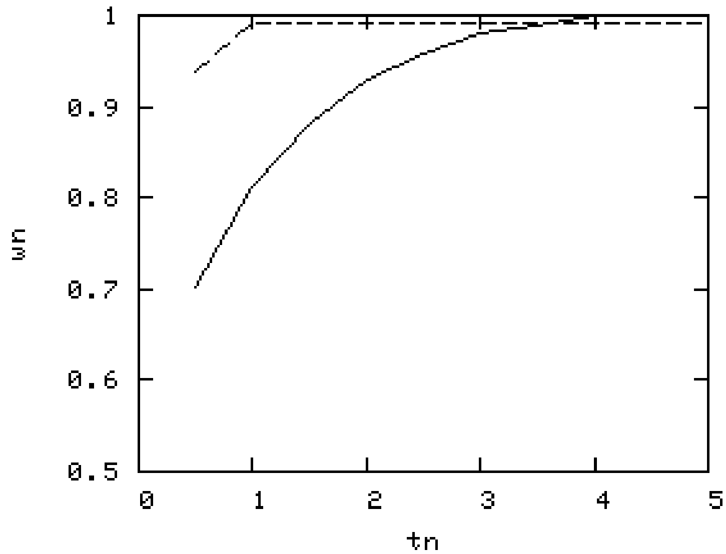

Figure 5: Actual vs. ideal

the initial population.

— With values of κ as small as .03 the GA convergence starts to improve, and the improvement continues up to values of .1 and more.

— For various combinations of other parameters, the improvement from $s = 16 \cdot l$ of larger multiples of l is often slight, and moderate at worst.

— Behavior improves from $l = 16$ to $l = 32$.

Figure 5 shows an example with $\lambda = .025$, $\kappa = .05$, $l = 32$, and for the actual GA (dashed lines) $s = 1024$.

References

[1] A. Benyi, S. Manago, A recursive formula for moments of a binomial distribution, *The College Mathematics Journal*, **36** (2005), 68-72.

[2] R. Cerf, Asymptotic convergence of a genetic algorithm, *Comptes Rendus de l'Academie des Sciences, Serie I: Mathematique*, **319** (1994), 271-276.

[3] D. Fogel, Asymptotic convergence properties of genetic algorithms and evolutionary programming: analysis and experiments, *Cybernetics and Systems*, **25** (1994), 389-407.

[4] D. Knuth, *The Art of Computer Programming I*, Addison-Wesley Publishing Company (1973).

[5] S. Louis, G. Rawlins, *Predicting Convergence Time for Genetic Algorithms*, Technical Report tr370, Department of Computer Science, Indiana University (1993).

[6] *Genetic Algorithm*, http://en.wikipedia.org/wiki/Genetic_algorithm.

[7] *Genetic Algorithms*, http://www.learnartificialneuralnetworks.com/geneticalg.html

[8] http://www.ks.uiuc.edu/Services/Class/PHYS498NSM/LectureNotes.html