Schema Theory

David White
Wesleyan University

November 30, 2009

Building Block Hypothesis

David White
Wesleyan University

Recall "Royal Roads" problem from September 21 class

Definition

Building blocks *are short groups of alleles that tend to endow chromosomes with higher fitness and are close to each other on the chromosome*

Theorem (Building Block Hypothesis)

Crossover benefits a GA by combining ever-larger hierarchical assemblages of building blocks.

Small BBs combine to create larger BB combinations, hopefully with high fitness. This is done in parallel. Recall that Random Mutation Hill Climber beat GA.

Questions

1. What laws describe the macroscopic behavior of GAs?
2. What predictions can be made about change in fitness over time?
3. How do selection, xover, and mutation affect this?
4. What performance criteria are appropriate for GAs?
5. When will a GA outperform hill climbers?

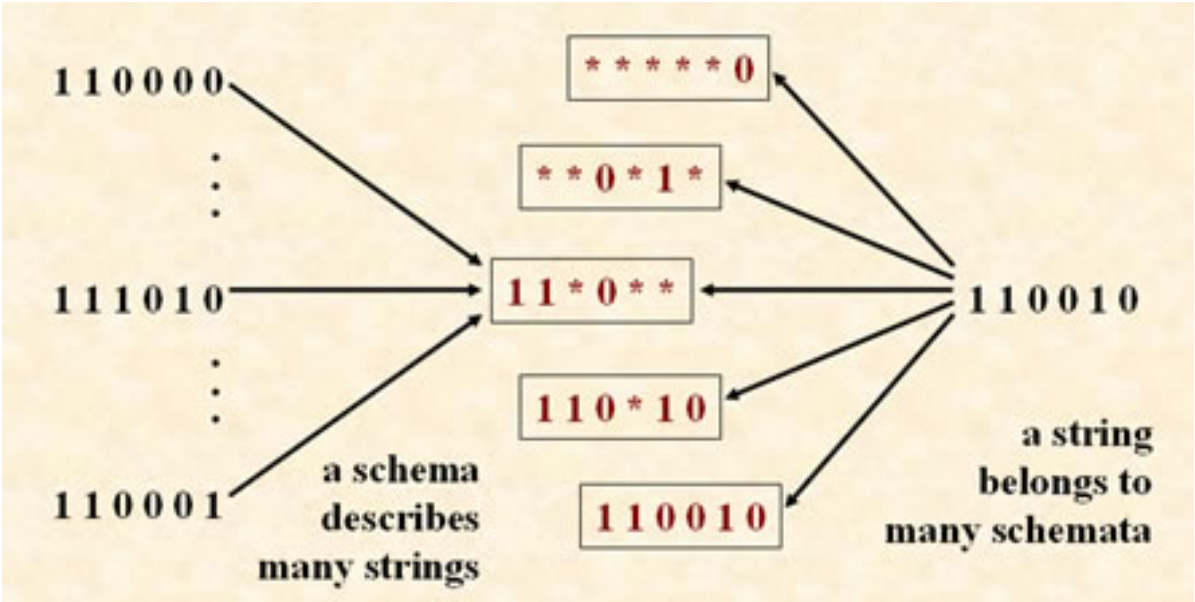
For simplicity assume a population of binary strings with one-point crossover and bit mutation.

Schema

Definition

*A **schema** is a string s from the alphabet $\{0, 1, *\}$*

s defines a hyperplane $H = \{t \mid t_i = s_i \text{ or } s_i = *\}$, also called a schema. H consists of length-l bit strings in the search space matching the s template



Idealized GA

On Royal Roads, **IGA** keeps one string with the best parts of all schemata and crosses it with new schema strings as they are found. It has indep. sampling in each schema

The IGA assumes prior knowledge of all schemata, which is not realistic. IGA works in parallel among schemata.

For N blocks of K ones each, IGA expected time is $O(2^K \ln N)$ whereas RMHC is $O(2^K N \ln N)$, proving GAs can beat RMHC. This is because doesn't need to do function evaluation, and IGA has no hitchhiking

GAs which approximate IGA can beat RMHC. They need:

1. Independent samples and slow convergence
2. Sequestering desired schemata
3. Fast xover with sequestered schemata
4. Large N so the factor in speed matters

Schema Theorem Idea

David White
Wesleyan University

GAs should identify, test, and incorporate structural properties hypothesized to give better performance

Schema formalize these structural properties
We can't see schemata in population, only strings

Definition

*The **fitness** of H is the average fitness of all strings in H.*

Estimate this with chromosomes in population matching s

Want: higher fitness schema get more chances to reproduce and GA balances exploration vs. exploitation

Two-Armed Bandit

David White
Wesleyan University

How much sampling should above-average schemata get?

On-line performance criterion: payoff at every trial counts in final evaluation. Need to find best option while maximizing overall payoff.

Gambler has N coins and a 2-armed slot machine with arm A_1 giving mean payoff μ_1 with variance σ_1^2, and same for A_2

Payoff processes are stationary and independent.

What strategy maximizes total payoff for $\mu_1 \geq \mu_2$?

$A_l(N, n)$ is arm with lower observed payoff (n trials)

$A_h(N, N - n)$ has higher observed payoff ($N - n$ trials)

Two-Armed Bandit Solution

$q = Pr(A_l(N,n) = A_1)$, $L(N-n,n) =$ losses over N trials

$L(N-n,n) = q \cdot (N-n) \cdot (\mu_1 - \mu_2) + (1-q) \cdot n \cdot (\mu_1 - \mu_2)$
(Probability of case) * (number of runs) * (payoff of case)

Maximize: $\frac{dL}{dn} = (\mu_1 - \mu_2)\left(1 - 2q + (N-2n)\frac{dq}{dn}\right) = 0$

$S = \Sigma$(payoffs of A_1-trials), $T = \Sigma$(payoffs of A_2-trials)

$q = P\left(\frac{S}{n} < \frac{T}{N-n}\right)$

Central Limit Theorem/Theory of Large Derivations:

$n^* \approx c_1 \ln\left(\frac{c_2 N^2}{\ln(c_3 N^2)}\right) \Rightarrow N - n^* \approx e^{cn^*}$

Do exponentially many more trials on current best arm

Two-Armed Bandit Interpretation

Similarly, schema theorem says instances of H in pop grow exponentially for high fitness, low length schemata H.

Direct analogy (GA schema are arms) fails because schema are not independent. Fix by partitioning search space into 2^k competing schema and running 2^k-armed bandit.

Best observed schema within a partition gets exponentially more samples than the next best.

Need uniform distribution of fitnesses for this argument.

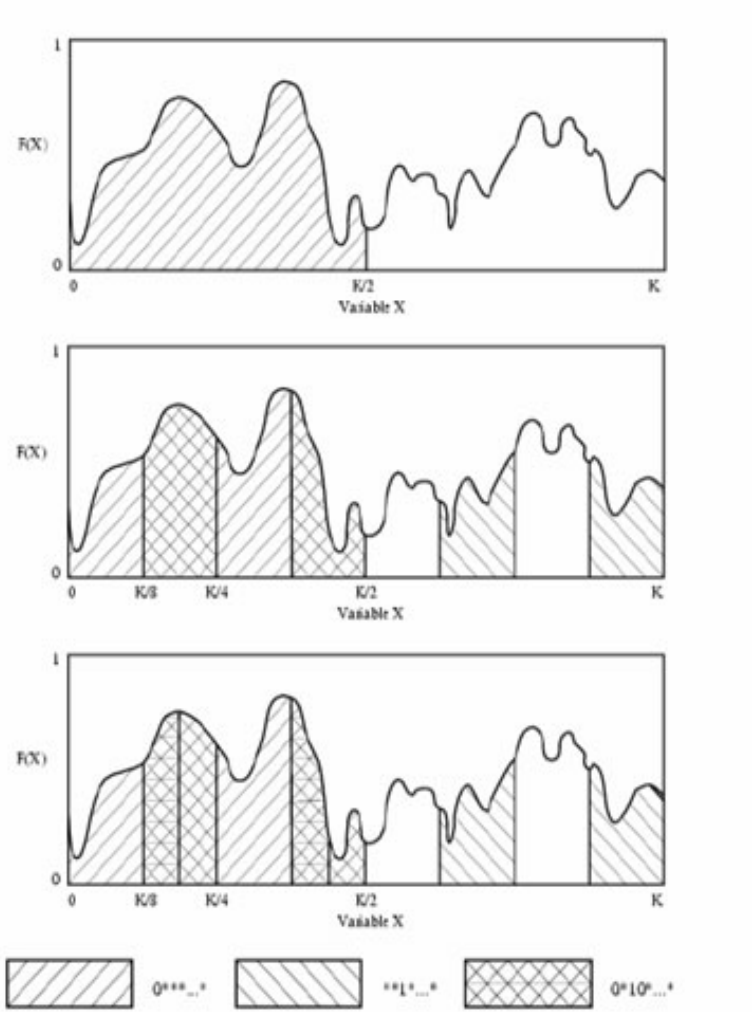
Biases introduced by selection mean static average fitness need not be correlated with observed average fitness.

Solution generalizes for 2^k-armed bandit

Hyperplane Partitions via Hashing

David White
Wesleyan University

Fitness vs. one variable as a $K = 4$-bit number



Order and Defining Length

Definition

*The **order** of a schema s is $o(s) = o(H) =$ the number of fixed positions (non-$*$) in s.*

Definition

*The **defining length** of a schema H is $d(H) =$ distance between the first and last fixed positions. Number of places where 1-point xover can distrupt s.*

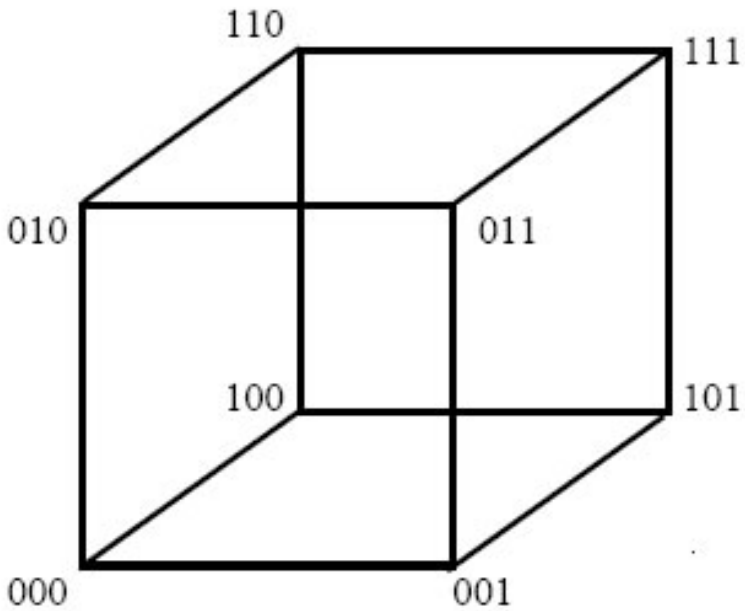
$O(10**0) = 3$, $d(1**0*1) = 5$, $d(*1*00) = 3$

A schema H matches $2^{l-o(H)}$ strings.
A string of length l is an instance of 2^l different schemata.

e.g. 11 is instance of $**, *1, 1*, 11$

Extended Example

Problem encoded with 3 bits has search space of size 8.
Think of this as a cube:



Corners - order 3, edges - order 2, faces - order 1
Hence the term “Hyperplane”

Implicit Parallelism

Not every subset of length l-bit strings can be described as a schema: only 3^l possible schemata

but 2^l strings of length $l \Rightarrow 2^{2^l}$ subsets of strings

Pop. of n strings has instances of between 2^l and $n \cdot 2^l$ diff. schemata. Each string gives info. on all schemata it matches.

Implicit parallelism: When GA evaluates fitness of pop. it implicitly evaluates fitness of many schema,

i.e. many hyperplanes are sampled and evaluated in an implicitly parallel fashion. We have parallelized our search of solution space.

Implicit Parallelism

Proposition (Implicit Parallelism)

A pop. of size n can process $O(n^3)$ schemata per generation.

i.e. these schemata are not disrupted by xover and mutation. Holds whenever $64 \leq n \leq 2^{20}$ and $l \geq 64$

$\phi =$ number of instances needed to process H. $\theta =$ highest order H-string in pop. Number of schema of order θ is $2^\theta \cdot \binom{l}{\theta} \geq n^3$ because $\theta = \log(n/\phi)$ and $n = 2^\theta \phi$

Note that small $d(H)$ schema are less likely to be disrupted by xover. A compact representation keeps alleles/loci together.

$S_c(H) = P(H \text{ survives under xover})$
$S_m(H) = P(H \text{ survives under mutation})$

Basic Schema Theorem

Assume fitness-proportional selection.
$\bar{f}(t)$ = average fitness of population at time t.
Expected number of offspring of string x is $f(x)/\bar{f}(t)$

$m(H,t)$ = the number of instances of H at time t

$\hat{u}(H,t) = \frac{\sum_{x \in H} f(x)}{m(H,t)}$ = observed ave. fitness at time t

Ignoring the effects of crossover and mutation:

$$E(m(H,t+1)) = \sum_{x \in H} \frac{f(x)}{\bar{f}(t)} = \frac{\hat{u}(H,t) m(H,t)}{\bar{f}(t)}$$

If $\hat{u}(H,t) = \bar{f}(t)(1+c)$ then $m(H,t) = m(H,0)(1+c)^t$
That is, above-average schemata grow exponentially

Factoring in xover and mutation

Each of the $o(H)$ fixed bits changes with probability p_m,
All stay unchanged with probability $S_m(H) = (1 - p_m)^{o(H)}$
To get a lower bound on $P(H$ destroyed by xover), assume xover within $d(H)$ is always disruptive.

$P(H$ destroyed$) \leq P($xover occurs within $d(H)) = p_c \left(\frac{d(H)}{l-1} \right)$

Thus, ignoring xover gains: $S_c(H) \geq 1 - p_c \left(\frac{d(H)}{l-1} \right)$

xover's reproduction helps schemata with higher fitness values. Both xover and mutation can create new instances of schema but it's unlikely. Both hurt long schemata more than short. Mutation gives diversity insurance

Inequalities assume independence of mutation b/t bits.

Schema Theorem

Theorem (Schema Theorem)

$$E(m(H,t+1)) \geq \frac{\hat{u}(H,t)}{\bar{f}(t)} m(H,t) \left(1 - p_c \frac{d(H)}{l-1}\right) (1-p_m)^{o(H)}$$

i.e. *short, low-order schemata with above average fitness (**building blocks**) will have exponentially many instances evaluated.* Theorem doesn't state how schema found

Parallels the Breeders Equation from quantitative genetics: $R = sh$ where R is the response to selection, s is the selection coefficient, and h is the heritability coefficient

Classical Version of Schema Theorem (less accurate):
$E(m(H,t+1)) \geq \frac{\hat{u}(H,t)}{\bar{f}(t)} m(H,t) \left(1 - p_c \frac{d(H)}{l-1} - p_m \cdot o(H)\right)$

Comes from $S_m(H) \geq (1 - o(H)p_m)$ when $p_m << 1$

BBH Revisited

BBH states that good GAs combine building blocks to form better solutions. BBH is unproven and criticized because it lacks theoretical justification. Evidence against:

1. Uniform outperformed one-point in Syswerda, but is very distruptive of short schemata.
2. Royal Roads
3. BBH is logically equivalent to some STRONG things

Some citing BBH are really assuming **SBBH** (Static BBH): Given any schema partition, a GA is expected to converge to the class with the best static average fitness

SBBH fails after convergence makes schemata samples not uniform (collateral convergence). It also fails when static average fitness has high variance.

Principle of Minimal Alphabets

David White
Wesleyan University

Holland argued for the optimality of binary encoding, using schema theory.

Implicit parallelism suggests we should try to maximize the number of schemata processed simultaneously

The number of possible schemata for alphabet A is $|A+1|^l$ Maximized when l is maximized. Amount of information needing to be stored is fixed, so l maximized when $|A|$ minimized

The smallest possible value of $|A|$ is 2, i.e. binary encoding

Counter-argument says $|A| > 2$ gives MORE hyperplane partitions, but independence may be lost. Issue is still unresolved.

Deception!

Deception occurs when low-order partitions contain misleading information about higher-order partitions.
e.g. strings of length $< L$ are winners iff they are all 1's but only the all 0 string is a winner of length L.

Fully deceptive: average schemata fitness indicate complement of global optimum is global optimum
Study of deception is concerned with function optimization

One solution if you have prior knowledge of the fitness function is to avoid deception via the encoding.

Some say deception is not a problem because the GA is a satisficer so it'll maximize cumulative payout regardless of deception or hidden fitness peaks

Examples show deception is neither necessary nor sufficient to cause GA difficulties

Is any of this useful?

Schema Theorem (ST) deals with expectation only and gives only a lower bound. P - problem, F - fix

P: Lower bound means it's impossible to use ST recursively to predict behavior over multiple generations

P: We can't say one presentation is better than another
F: Formulate Exact ST (EST) as on the coming slides. This gives one criteria for comparing performance

P: ST fails in the presence of noise/stochastic effects.

F: Poli reinterpreted EST as a conditional statement about random variables (Conditional ST = CST) which estimates the expected proportion of a schema.

Schema Theorems without Expectation

David White
Wesleyan University

$\alpha = P(H$ survives or is created after variation),
$k > 0$ any constant, $\mu = n\alpha$, $\sigma^2 = n\alpha(1-\alpha)$

Theorem (Two-sided probabilistic Schema Theorem)

$$P(|m(H,t+1) - n\alpha| \leq k\sqrt{n\alpha(1-\alpha)}) \geq 1 - 1/k^2$$

This is Chebychev's Inequality: $P(|X - \mu| < k\sigma) \geq 1 - 1/k^2$

Theorem (Probabilistic Schema Theorem)

$$P(m(H,t+1) > n\alpha - k\sqrt{n\alpha(1-\alpha)}) \geq 1 - 1/k^2$$

This theorem lets you predict the past from the future.
Also, discovering one bit of the solution per generation lets us recursively apply CST to find conditions on the inital population under which the GA will converge
This assumes we know BBs and fitnesses in pop.

Is any of this useful?

P: ST assumed bit-string representation, 1-pt xover, etc
F: MANY papers generalize ST and EST to other GAs, context-free grammars, and GP (many versions)

P: EST expresses $E(m(H, t+1))$ as a function of microscopic quantities (properties of individuals) rather than macroscopic quantities (properties of schemata).

F: Riccardo Poli reformulated EST to fix this

Schema Theory gives a theoretical basis for why GAs work, tells us about GA convergence, and applies to EC broadly

EST needs infinite population for true accuracy, but approximations to finite populations exist.

Exact Model

David White
Wesleyan University

Assume FPS, bit mutation, and 1-pt xover creating only one child per generation. So n recombinations needed per gen.

$p_i(t) =$ proportion of pop. in gen. t matching string i.
$s_i(t) =$ prob. that an instance of string i will be selected as a parent. In generation t, $p(t)$ is composition of pop. and $s(t)$ is selection probabilities

We define an operator G s.t. $Gs(t) = s(t+1) = G^{t+1}s(0)$
Fitness matrix $F =$ diagonal matrix with $F_{i,i} = f(i)$
M covers xover and mutation, with $M_{i,j} = r_{i,j}(0)$
$r_{i,j}(k) = P(k$ produced from recombination of i and $j)$

From M define T to cover $r_{i,j}(k)$'s and then $G = F \circ T$
Define $G_p(x) = T(Fx/|Fx|)$ where $|v| = \Sigma$(components)
$Gs = ks(t+1), G_p(p(t)) = p(t+1)$ as $n \to \infty$

Proof Sketch

$r_{i,j}(0) = r^m_{i,j}(0) \cdot r^C_{i,j}(0)$ for mut. and xover factors

$P(i \text{ mutated to all 0s}) = p_m^{|i|}(1-p_m)^{l-|i|}$ so
$r^m_{i,j}(0) = \frac{1}{2}(1-p_c)[p_m^{|i|}(1-p_m)^{l-|i|} + p_m^{|j|}(1-p_m)^{l-|j|}]$

$P(\text{xover at } c) = 1/(l-1)$; h, k are offspring.

$$r^C_{i,j}(0) = \frac{1}{2}\frac{p_c}{l-1}\sum_{c=1}^{l-1}[p_m^{|h|}(1-p_m)^{l-|h|} + p_m^{|k|}(1-p_m)^{l-|k|}]$$

$i_1 = \text{substring}(i, 0, l-c-1)$, $i_2 = \text{substring}(i, l-c-1, c)$
Clever Trick: $|i_2| = |(2^c - 1) \wedge i|$ for $\wedge =$ "and"

Use logical operators and permutations to get T from M

Dynamics

Iterating G forms a dynamical system on $\{s\text{-vectors}\}$

$F(s(t)) = s(t)$ iff $s =$ maximally fit limit of pop.

$M(s(t)) = s(t)$ iff s_i constant over all i

F focusing vs. M diffusing explains **punctuated equilibria** - long stability then quick rises in fitness

Problem: this needs infinite population, and expected proportions are not always met due to **sampling error**

Solution: **Markov Chains** - stochastic processes where $P(\text{state } j \text{ at time } t)$ depends only time $t-1$ state

Finite Population Model

David White
Wesleyan University

State of GA is population at time t. Set of all states is set of possible populations of size n (matrix Z).

$Z_{y,i} = \#$ occurrences of string y in i-th population ϕ_i

$N = \binom{n+2^l-1}{2^l-1}$ possible populations of size n

$\frac{n!}{Z_{0,j}!Z_{1,j}!\cdots Z_{2^l-1,j}!}$ ways to form pop. P_j

$$\text{Markov Transition Matrix has } Q_{i,j} = n! \prod_{y=0}^{2^l-1} \frac{p_i(y)^{Z_{y,j}}}{Z_{y,j}!}$$

$$\text{Substitute in } p_i(y) = \left[T\left(\frac{F\phi_i}{|F\phi_i|} \right) \right]_y$$

Final Results

As $n \to \infty$, Markov trajectories converge to iterates of G (or G_p) with probability arb. close to 1.
For large n infinite model mimics finite model. As $n \to \infty$ the time GA spends away from G_p fixed points goes to 0

Conjecture (Vose's Conjecture)

Short-term GA behavior is determined by initial pop. but long-term behavior is determined only by "GA surface" where population trajectories occur.

Supported by some simulation evidence, but it suggests every simple GA converges to a unique steady state distribution. Suzuki (1995) found a counter-example.

Problem: The matrix Z is HUGE. Solution is to use statistical mechanics to get at GA behavior using macroscopic statistics. Poli's work applies

References

1. www.cse.unr.edu/ sushil/class/gas/notes/ schemaTheoremSushil.ppt
2. www.cs.utk.edu/ mclennan/Classes/420/ handouts/Part-5A.ppt
3. www.egr.msu.edu/ goodman/ PradeepClassGoodmanGATutorial.ppt
4. http://www.cs.bris.ac.uk/Teaching/Resources/ COMSM0302/lectures/schema_theory08.p.pdf
5. Mitchell, M. *An Introduction to Genetic Algorithms*
6. Syswerda, G. "Uniform crossover in genetic algorithms" in Proceedings of the Third International Conference on GA.
7. Burjorjee, K. "The Fundamental Problem with the Building Block Hypothesis"

For papers, see http://cswww.essex.ac.uk/staff/poli/papers/publications.html