Dynamical Systems Model of the Simple Genetic Algorithm

Introduction to Michael Vose's Theory

Summer Lecture Series 2002

Rafal Kicinger
rkicinge@gmu.edu

Overview

- Introduction to Vose's Model
- Defining Mixing Matrices
- Finite Populations
- Conclusions

Overview

- Introduction to Vose's Model
  - SGA as a Dynamical System
  - Representing Populations
  - Random Heuristic Search
  - Interpretations and Properties of $\mathcal{G}(x)$
  - Modeling Proportional Selection
- Defining Mixing Matrices
- Finite Populations
- Conclusions

Overview

- Introduction to Vose's Model
- Defining Mixing Matrices
  - What is Mixing?
  - Modeling Mutation
  - Modeling Recombination
  - Properties of Mixing
- Finite Populations
- Conclusions

Overview

- Introduction to Vose's Model
- Defining Mixing Matrices
- Finite Populations
  - Fixed-Points
  - Markov Chain
  - Metastable States
- Conclusions

Overview

- Introduction to Vose's Model
- Defining Mixing Matrices
- Finite Populations
- Conclusions
  - Properties and Conjectures of $\mathbf{G}(x)$
  - Summary

Overview

- Introduction to Vose's Model
  - SGA as a Dynamical System
  - Representing Populations
  - Random Heuristic Search
  - Interpretations and Properties of $\mathbf{G}(x)$
  - Modeling Proportional Selection
- Defining Mixing Matrices
- Finite Populations
- Conclusions

SGA as a Dynamical System

What is a dynamical system?

$\rightarrow$ a set of possible states, together with a rule that determines the present state in terms of past states.

When a dynamical system is deterministic?

$\rightarrow$ If the present state can be determined *uniquely* from the past states (no randomness is allowed).

SGA as a Dynamical System

1. SGA usually starts with a random population.
2. One generation later we will have a new population.
3. Because the genetic operators have a random element, we cannot say exactly what the next population will be (algorithm is not deterministic!!!).

SGA as a Dynamical System

However, we can calculate:

→ the probability distribution over the set of possible populations defined by the genetic operators

→ expected next population

As the population size tends to infinity:

→ the probability that the next population will be the expected one tends to 1 (algorithm becomes deterministic)

→ and the trajectory of expected next population gives the actual behavior.

Representing Populations

Let Z represent a search space containing s elements,

$$Z = \{z_0, z_1, \ldots, z_{s-1}\}$$

Example:

Search space of fixed-length binary strings of length $l=2$. Then,

z_0=00 z_1=01 z_2=10 z_3=11

The size of the search space is given by $s=2^l$

Representing Populations

Population p is a point in the space of all possible populations.

We can represent a population p by considering the number of copies a_k of each element z_k that p contains as a fraction of the total population size r, that is:

$$p_k = \frac{a_k}{r}$$

This gives us a vector $p=(p_0,p_1,\ldots p_{s-1})$

Representing Populations

Example cont. ($l=2$):

Suppose that a population consists of:

{00,00,01,10,10,10,10,10,11,11}

Then r =10 and p=(0.2,0.1,0.5,0.2)

Representing Populations

Properties of population vectors:

1. p is an element of the vector space R^s (addition and/or multiplication by scalar produce other vectors within R^s)
2. Each entry p_k must lie in the range [0,1]
3. All entries of p sum to 1

The set of all vectors in R^s that satisfy these properties is called the simplex and denoted by Λ.

Representing Populations

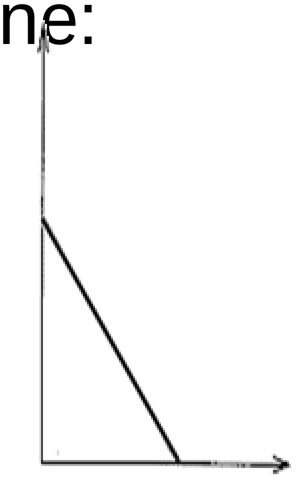
Examples of Simplex Structures:

1. The simplest case:

   Search space has only two elements
   $Z = \{z_0, z_1\}$

Population vectors are contained in R^2

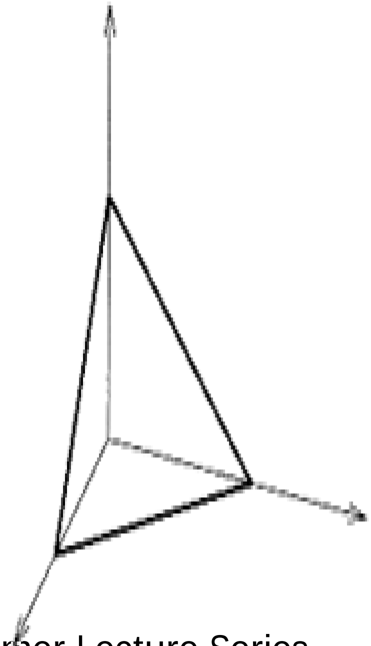
Simplex Λ is a segment of a straight line:



Representing Populations

2. Search space Z has 3 elements, $Z=\{z_0,z_1,z_2\}$

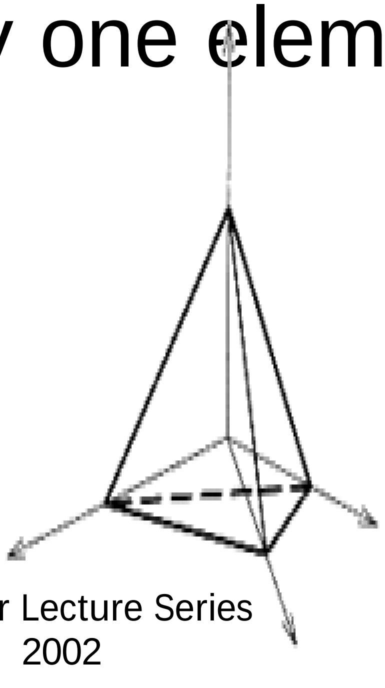
   Simplex Λ is now a triangle with vertices at (1,0,0), (0,1,0), (0,0,1).



Representing Populations

In general, in ***s*** dimensional space the simplex forms ***(s-1)***-dimensional object (a hyper-tetrahedron).

The vertices of the simplex correspond to populations with copies of only one element.



Representing Populations

Properties of the Simplex:

$\rightarrow$ Set of possible populations of a given size r takes up a finite subset of the simplex.

$\rightarrow$ Thus, the simplex contains some vectors that could never be real populations because they have irrational entries.

$\rightarrow$ But, as the population size r tends to infinity, the set of possible populations becomes dense in the simplex.

Random Heuristic Search

Algorithm is defined by a “heuristic function”

$$\boldsymbol{G}(x) = \Lambda \rightarrow \Lambda$$

```
1. Let $x$ be a random population of size $r$
2. $y$ <- $0$ Î $R^s$
3. FOR $i$ from $1$ to $r$ DO
4.     Choose $k$ from the probability
           distribution $\boldsymbol{G}(x)$
5.     $y$ <- $y + 1/r \times e_k$ (add $k$ to population $y$)
6. ENDFOR
```

Interpretations of $\mathbf{G}(x)$

1. $\mathbf{G}(x)$ is the expected next generation population
2. $\mathbf{G}(x)$ is the limiting next population as the population size goes to infinity
3. $\mathbf{G}(x)_j$ is the probability that j Î Z is selected to be in the next generation

Introduction to Vose's Dynamical Systems Model

Properties of $\mathcal{G}(x)$

$\boldsymbol{G}(x) = \boldsymbol{U}(\boldsymbol{C}(\boldsymbol{F}(x)))$, where $\boldsymbol{F}$ describes selection, $\boldsymbol{U}$ describes mutation, and $\boldsymbol{C}$ describes recombination.

$x \rightarrow \boldsymbol{G}(x)$ is a discrete-time dynamical system

Simple Genetic Algorithm

1. Let X be a random population of size r.
2. To generate a new population Y do the following r times:

   - choose two parents from X with probability in proportion to fitness

   - apply crossover to parents to obtain a child individual

   - apply mutation to the child

   - add the child to new population y
3. Replace X by Y
4. Go to step 2.

Modeling Proportional Selection

Let $p=(p_0,p_1,\ldots p_{s-1})$ be our current population.

We want to calculate the probability that z_k will be selected for the next population.

Using fitness proportional selection, we know this probability is equal to:

$$\frac{f(z_k)\cdot p_k}{\overline{f}(p)}$$

Modeling Proportional Selection

The average fitness of the population p can be calculated by:

$$\overline{f}(p) = \sum_{k=0}^{s-1} f(z_k) \cdot p_k$$

We can create a new vector q, where q_k equals the probability that z_k is selected.

We can think of q as a result of applying an operator $\boldsymbol{F}$ to p, that is $q = \boldsymbol{F}\, p$

Modeling Proportional Selection

Let S be a diagonal matrix S such that:

$$S_{k,k}=f(z_k)$$

Then we can use the following concise formula for q:

$$q = \boldsymbol{F}\, p = \frac{1}{\overline{f}(p)} \cdot Sp$$

Modeling Proportional Selection

Probabilities in q define the probability distribution for the next population, if only selection is applied.

This distribution specified by the probabilities $q_0,\ldots,q_{s\text{-}1}$ is a multinomial distribution.

Modeling Proportional Selection

Example:

Let $Z=\{0,1,2\}$

Let $f=(3,1,5)^T$

Let $p=(\frac{1}{4},\frac{1}{2},\frac{1}{4})^T$

$\overline{f}(p)=3\times\frac{1}{4}+1\times\frac{1}{2}+5\times\frac{1}{4}=5/2$

$q = \boldsymbol{F}\,p=$

$$\frac{1}{\overline{f}(p)}\cdot Sp=\frac{1}{\frac{5}{2}}\cdot\begin{bmatrix}3&0&0\\0&1&0\\0&0&5\end{bmatrix}\begin{bmatrix}\frac{1}{4}\\\frac{1}{2}\\\frac{1}{4}\end{bmatrix}=\begin{bmatrix}\frac{3}{10}\\\frac{1}{5}\\\frac{1}{2}\end{bmatrix}$$

Modeling Proportional Selection

If there is a unique element z_k of maximum fitness in population p, then the sequence p, $\boldsymbol{F}(p), \boldsymbol{F}(\boldsymbol{F}(p))$, …converges to the population consisting only of z_k, which is the unit vector e_k in R^s.

Thus, repeated application of selection operator $\boldsymbol{F}$ will lead the sequence to a fixed-point which is a population consisting only of copies of the element with the highest fitness from the initial population.

Overview

- Introduction to Vose's Model ☑
- Defining Mixing Matrices ←
  - What is Mixing?
  - Modeling Mutation
  - Modeling Recombination
  - Properties of Mixing
- Finite Populations
- Conclusions

Defining Mixing Matrices

What is Mixing?

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

Defining Mixing Matrices

Modeling Mutation

We want to know the probability that after mutating individuals that have been selected, we end up with a particular individual.

There are two ways to obtain copies of z_i after mutation:

- other individual z_j is selected and mutated to
  produce z_i

- z_i is selected itself and not mutated

Defining Mixing Matrices

Modeling Mutation

The probability of ending up with z_i after selection and mutation is:

$$\sum_{j=0}^{s-1} U_{i,j} q_j$$

where $U_{i,j}$ is the probability that z_j mutates to form z_i

Example:

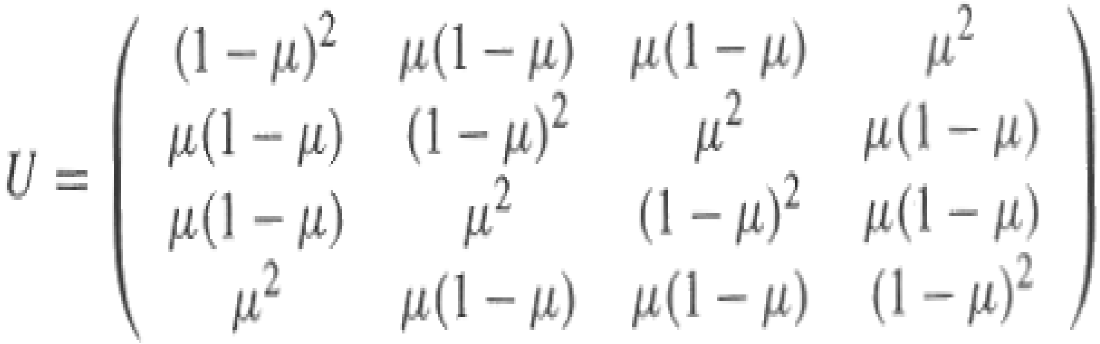
The probability of mutating z_5=101 to z_0=000 is equal to:

$U_{0,5}=m^2(1- m)$

Defining Mixing Matrices

Modeling Mutation

We can put all the $U_{i,j}$ probabilities in the matrix U. For example, in case of $l=2$ we obtain:



$$U = \begin{pmatrix} (1-\mu)^2 & \mu(1-\mu) & \mu(1-\mu) & \mu^2 \\ \mu(1-\mu) & (1-\mu)^2 & \mu^2 & \mu(1-\mu) \\ \mu(1-\mu) & \mu^2 & (1-\mu)^2 & \mu(1-\mu) \\ \mu^2 & \mu(1-\mu) & \mu(1-\mu) & (1-\mu)^2 \end{pmatrix}$$

Defining Mixing Matrices

Modeling Mutation

If p is a population, then $(\boldsymbol{U}p)_j$ is the probability that individual j results from applying only mutation to p.

With a positive mutation rate less than 1, the sequence p, $\boldsymbol{U}(x)$, $\boldsymbol{U}(\boldsymbol{U}(x))$, ... converges to the population with all elements of Z represented equally (the center of the simplex).

Defining Mixing Matrices

Modeling Mutation

The probability of ending up with z_i after applying mutation and selection can be represented as the one time-step equation:

$$p(t+1)=\boldsymbol{U} \circ \boldsymbol{F}\, p(t)=$$

$$\frac{1}{\overline{f}(p)} U\ Sp(t)$$

Defining Mixing Matrices

Modeling Mutation

Will this sequence converge as time goes to infinity?

This sequence will converge to a fixed-point p satisfying:

$$\boldsymbol{U}\, S\, p = \bar{f}(p)\, p$$

This equation states that the fixed-point population p is an eigenvector of the matrix $\boldsymbol{U}\, S$ and that the average fitness of p is the corresponding eigenvalue.

Defining Mixing Matrices

Modeling Mutation

Perron-Frobenius Theorem (for matrices with positive real entries)

From this theorem we know that $\boldsymbol{U}\,S$ will have exactly one eigenvector in the simplex, and that this eigenvector corresponds to the leading eigenvalue (the one with the largest absolute value).

Defining Mixing Matrices

Modeling Mutation

Summarizing, for SGA under proportional selection and bitwise mutation:

1. Fixed-points are eigenvectors of $\boldsymbol{U}$S, once they have been scaled so that their components sum to 1.
2. Eigenvalues of $\boldsymbol{U}$S give the average fitness of the corresponding fixed-point populations.
3. Exactly one eigenvector of US is in the simplex Λ.
4. This eigenvector corresponds to the leading eigenvalue.

Modeling Recombination

Effects of applying crossover can be represented as an operator $\boldsymbol{C}$ acting upon simplex Λ.

$(\boldsymbol{C}\,p)_k$ gives the probability of producing individual z_k in the next generation by applying crossover.

Modeling Recombination

Let ⊕ denote bitwise mod 2 addition (XOR)

Let ⊗ denote bitwise mod 2 multiplication (AND).

If $m \in Z$, let m denote the ones complement of m.

Example:

Parent 1:	01010010101 = z_i
Parent 2:	11001001110 = z_j
Mask:	11111100000 = m
Child:	01010001110 = z_k

Modeling Recombination

$z_k = (z_i \otimes m) \oplus \overline{(z}_j \otimes m)$

Let $r(i,j,k)$ denote the probability of recombining i and j and obtaining k.

Let C_0 be a $s \times s$ matrix defined by:

$C_{i,j}=r(i,j,0)$

Let s_k be the permutation matrix so that

$s_k\, e_i=e_{i\oplus k}$ where e_i is the i-th unit vector

Modeling Recombination

Define $\boldsymbol{C}$: $\Lambda \rightarrow \Lambda$ by

$$\boldsymbol{C}(p) = (s_k\, p)^T C_0 (s_k\, p)$$

Then $\boldsymbol{C}$ defines the effect of recombination on a population p.

Defining Mixing Matrices

Modeling Recombination

Example (from Wright):

$l=2$ binary strings

String	Fitness
00	3
01	1
10	2
11	4

Defining Mixing Matrices

Modeling Recombination

Assume an initial population vector of $p=(1/4, 1/4, 1/4, 1/4)^T$

$q= \boldsymbol{F}(p)= \left\langle \frac{3}{10}, \frac{1}{10}, \frac{2}{10}, \frac{4}{10} \right\rangle^T$

Assume one-point crossover with crossover rate of ½

$$C_0 = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ \frac{1}{4} & 0 & 0 & 0 \end{bmatrix}$$

Modeling Recombination

For example, the third component of $\boldsymbol{C}(q)$ is computed by:

$\boldsymbol{C}(q)_2 =$

$$p^T \qquad \boldsymbol{s}_2^{\,T} \qquad C_0 \qquad \boldsymbol{s}_2 \qquad p$$

$$\begin{bmatrix} \frac{3}{10} & \frac{1}{10} & \frac{2}{10} & \frac{4}{10} \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ \frac{1}{2} & \frac{1}{4} & 0 & 0 \\ \frac{1}{4} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{3}{10} \\ \frac{1}{10} \\ \frac{2}{10} \\ \frac{4}{10} \end{bmatrix} = \frac{5}{20}$$

Modeling Recombination

Similarly we can calculate other components and finally obtain:

$\boldsymbol{C}(q) = \left\langle \frac{5}{20}, \frac{3}{20}, \frac{5}{20}, \frac{7}{20} \right\rangle^T$

Now after applying mutation operator with mutation rate of 1/8 and we get:

$$\mathcal{U}\left(\left\langle \frac{5}{20}, \frac{3}{20}, \frac{5}{20}, \frac{7}{20} \right\rangle^T\right) = \begin{bmatrix} \frac{49}{64} & \frac{7}{64} & \frac{7}{64} & \frac{1}{64} \\ \frac{7}{64} & \frac{49}{64} & \frac{1}{64} & \frac{7}{64} \\ \frac{7}{64} & \frac{1}{64} & \frac{49}{64} & \frac{7}{64} \\ \frac{1}{64} & \frac{7}{64} & \frac{7}{64} & \frac{7}{64} \end{bmatrix} \begin{bmatrix} \frac{5}{20} \\ \frac{3}{20} \\ \frac{5}{20} \\ \frac{7}{20} \end{bmatrix} = \begin{bmatrix} \frac{77}{320} \\ \frac{59}{320} \\ \frac{83}{320} \\ \frac{101}{320} \end{bmatrix}$$

Defining Mixing Matrices

Properties of Mixing

For all the usual kinds of crossover that are used in GAs, the order of crossover and mutation doesn’t matter.

$$\boldsymbol{U} \circ \boldsymbol{C} = \boldsymbol{C} \circ \boldsymbol{U}$$

The probability of creating a particular individual is the same.

Defining Mixing Matrices

Properties of Mixing

This combination of crossover and mutation (in either order) gives the mixing scheme for the GA, denoted by $\boldsymbol{M}$.

$$\boldsymbol{M} = \boldsymbol{U} \circ \boldsymbol{C} = \boldsymbol{C} \circ \boldsymbol{U}$$

The k-th component of $\boldsymbol{M}$ p is:

$$\boldsymbol{M}(p)_k = \boldsymbol{C}(\boldsymbol{U}\, p)_k = (\boldsymbol{U}\, p)^T \cdot (\boldsymbol{C}_k \boldsymbol{U}\, p)$$

Defining Mixing Matrices

Properties of Mixing

Let us define $M_k = U\,C_k\,U$

The (i,j)th entry of M_k is the probability that z_i and z_j, after being mutated and recombined, produce z_k.

Then the mixing scheme is given by:

$$\boldsymbol{M}(p)_k = p^T \cdot (M_k\, p) = (s_k\, p)^T \cdot (M_0\, s_k\, p)$$

All the information about mutating and recombining is held in the matrix M_0 called the mixing matrix.

Overview

- Introduction to Vose's Model ☑
- Defining Mixing Matrices ☑
- Finite Populations ←
  - Fixed-Points
  - Markov Chain
  - Metastable States
- Conclusions

Finite Populations

Fixed-Points

If the population size r is finite, then each component p_i of a population vector p must be a rational number with r as a denominator.

The set of possible finite populations of size r forms a discrete lattice within the simplex Λ.

Finite Populations

Fixed-Points

Consequence:

Fixed-point population described by the infinite population model might not actually exist as a possible population!!!

Finite Populations

Markov Chain

Given an actual (finite) population represented by the vector $p(t)$, we have a probability distribution over all possible next populations defined by $G(p)=p(t+1)$.

The probability of getting a particular population depends only on the previous generation → Markov Chain.

Finite Populations

Markov Chain

A Markov Chain is described by its transition matrix Q.

$$Q_{q,p} = r!\prod_{j=0}^{s-1}\frac{(G(p)_j)^{rq_j}}{(rq_j)!}$$

$Q_{q,p}$ is the probability of going from population p to population q.

Finite Populations

Markov Chain

→ *$p(t+1)$* itself might not be an actual population

→ *$p(t+1)$* is the expected next population

→ Can think of the probability distribution clustered around that population

→ Populations that are close to it in the simplex will be more likely to occur as a next population than the ones that are far away

Finite Populations

Markov Chain

- → A good way to visualize this is to think of the operator G as defining an arrow at each point in the simplex
- → At a fixed-point of G, the arrow has 0 length
- → Thus, SGA is likely to spend much of its time at populations that are in the vicinity of the infinite population fixed-point

Finite Populations

Metastable States

Metastable states are parts of the simplex where the force of G is small, even if these areas are not near the fixed-point.

They are important in understanding the long-term behavior of a finite population GA.

Finite Populations

Metastable States

We extend G to apply to the whole of R^s.

Perron-Frobenius theory predicts only one fixed-point in the simplex, but we are now considering the action of G on the whole of R^s.

If there are other fixed-point close to the simplex, then by continuity of G, there will be a metastable region in that part of the simplex.

Finite Populations

Metastable States

Metastable states are simply other eigenvectors of U S suitably scaled so that their components sum to one.

To find potential metastable states within the simplex, we simply calculate all the eigenvectors of US

Overview

- Introduction to Vose's Model ☑
- Defining Mixing Matrices ☑
- Finite Populations ☑
- Conclusions
  - Properties and Conjectures of $\boldsymbol{G}(x)$ ←
  - Summary

Conclusions

Properties and Conjectures of $\mathbf{G}(x)$

The principle conjecture:

$\mathbf{G}$ is focused under reasonable assumptions about crossover and mutation

→ Known to be true if mutation is defined bitwise with a mutation rate <0.5 and there is no crossover.

→ When there is crossover it is known to be true when the fitness function is linear (or near to linear) and the mutation rate is small.

Conclusions

Properties and Conjectures of $\boldsymbol{G}(x)$

The second conjecture:

Fixed points of $\boldsymbol{G}$ are hyperbolic for nearly all fitness functions

→ Important for determining the stability of fixed points

→ Known to be true for the case of fixed-length binary strings, proportional selection, any kind of crossover, and mutation defined bitwise with a positive mutation rate

Properties and Conjectures of $\boldsymbol{G}\,(x)$

The third conjecture:

$\boldsymbol{G}$ is well-behaved

→ Known to be true if the mutation rate is positive but < 0.5 and if crossover is applied at a rate that is less than 1.

Properties and Conjectures of $\mathbf{G}(x)$

Assuming all three conjectures are true, then the following properties follow:

1. There are only finitely many fixed-points of $\mathbf{G}$.
2. The probability of picking a population p, such that iterates of $\mathbf{G}$ applied to p converge on an unstable fixed-point in zero.
3. The infinite population GA converges to a fixed-point in logarithmic time.

Conclusions

Summary

Michael Vose’s theory of the SGA:

→ Gives a general mathematical framework for the analysis of the SGA

→ Uses dynamical systems models to predict the actual behavior (trajectory) of the SGA

→ Provides results that are general in nature, but also applicable to real situations

→ Lays some theoretical foundations toward building the GA theory

Conclusions

Summary

But...

→ Is intractable in all except for the simple cases

→ Approximations are necessary to the Vose SGA model to make it tractable in real situations

Overview

- Introduction to Vose's Model ☑
- Defining Mixing Matrices ☑
- Finite Populations ☑
- Conclusions ☑