Analysis and Simulation Of A Simple Evolutionary System

Mahendra Duwal Shrestha

Overview

Overview

Research Question 3

Violation in Mutation
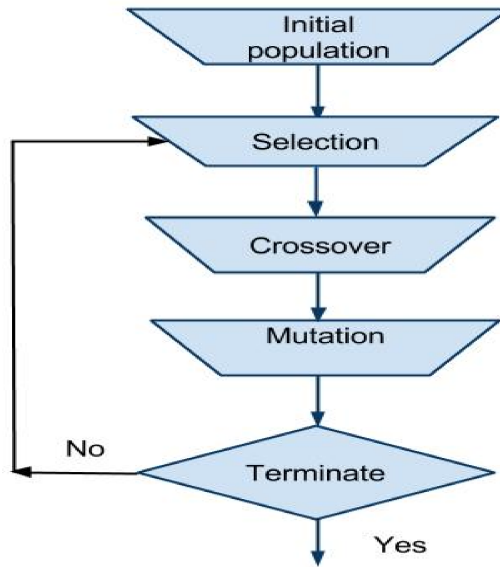
Simulation and Results

Conclusion

Research Question 4

Violation in Crossover

Simulation and Results

Conclusion

Finite Population GA

Population P, size *r*, length *l*

Population members represented by binary strings (10010011)

Two parents u and v selected to produce two offsprings u' and v' through crossover and mutation

Keep one of u' and v' with uniform probability

Replace P by new generations formed after *r* offspring are generated

Process is repeated until system stops to improve or threshold is reached

Infinite Population Model

Population modeled as vector p

p_j is the proportion of string j in the population

G is the function mapping infinite population p to the next generation

$G(p)$ is probability vector such that

$\quad G(p)_j =$ the probability that string j occurs in the next generation

The evolution of infinite population p is the sequence

$\quad p \rightarrow G(p) \rightarrow G(G(p)) \rightarrow \cdots$

Background

Several people working in the 1950s and the 1960s – like Box (1957), Friedman(1959), Bledsoe (1961), Bremermann (1962), and Reed, Toombs and Baricelli (1967) in evolution-inspired algorithms

Holland and colleagues formalized and promoted population based algorithms with crossover and mutation

In 1991, Vose and Liepins introduced model to compute infinite population trajectories

Given a finite population represented by vector $\boldsymbol{p}$, the infinite population model computes the expected proportion $G(\boldsymbol{p})_i$ of string i in the next generation

Nix and Vose explored relationship of finite population GA and infinite population model

They proved the short term trajectory followed by a finite population is related to the evolutionary path determined by the infinite population model

τ is stochastic transition rule that maps P to P'

p is population vector corresponding to population P (P chosen from some search space Ω)

Given population size r, a population vector *p* determines a population

The next population vector $\tau(p)$ cannot be predicted by certainty

G is a function that maps *p* to a vector whose ith component is the probability that the ith element of Ω is chosen

$G(p)$ specifies distribution of population in the next generation

$$
\begin{aligned}
Q_{p,q} &= r! \prod \frac{(\mathcal{G}(p)_j)^{rq_j}}{(rq_j)!} \\
&= \exp\{-r \sum q_j \log \frac{q_j}{\mathcal{G}(p)_j} - \sum (\log \sqrt{2\pi r q_j} + \frac{1}{12 r q_j + \theta(r q_j)}) \qquad (1.1) \\
&\quad + O(\log r)\}
\end{aligned}
$$

θ is a function such that $0 < \theta < 1$

Random Heuristic Search

Finite population sequence $\boldsymbol{p}$, $\tau(\boldsymbol{p})$, $\tau^2(\boldsymbol{p})$, ... forms Markov chain with transition matrix $Q_{\boldsymbol{p},\boldsymbol{q}}$

$G(\boldsymbol{p})$ is the expected next generation $\mathcal{E}(\tau(\boldsymbol{p}))$

$\sum q_j \log \dfrac{q_j}{\mathcal{G}(\boldsymbol{p})!}$ denotes discrepancy of $\boldsymbol{q}$ (how far $\boldsymbol{q}$ is from $G(\boldsymbol{p})$)

$\exp\{-r \sum q_j \log \dfrac{q_j}{\mathcal{G}(\boldsymbol{p})_j}\}$ indicates probability that $\boldsymbol{q}$ is next generation decreases exponentially with constant r as the discrepancy increases

$\sum (\log \sqrt{2\pi r q_j} + \dfrac{1}{12 r q_j + \theta(r q_j)})$ measures dispersion of $\boldsymbol{q}$

$\exp\{-\sum (\log \sqrt{2\pi r q_j} + \dfrac{1}{12 r q_j + \theta(r q_j)})\}$ indicates the probability that $\boldsymbol{q}$ is the next generation decays exponentially with increasing dispersion
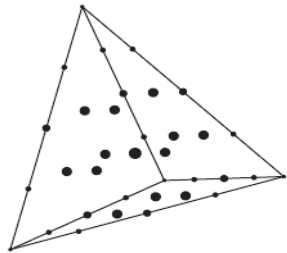
Figure 1.2: Population points

Finite populations are represented by dots
Smaller dots, lesser dispersion
Larger dots, higher dispersion

The variance of the next generation (with respect to the expected population) is

$$\mathcal{E}(\|\tau(\boldsymbol{p}) - \mathcal{G}(\boldsymbol{p})\|^2) = \frac{1 - \|\mathcal{G}(\boldsymbol{p})\|^2}{r} \qquad (1.2)$$

Research Question 1:

Distance Between Finite Population and Infinite Population

$$\mathcal{E}(\|\tau(\boldsymbol{p}) - \mathcal{G}(\boldsymbol{p})\|^2) = \frac{1 - \|\mathcal{G}(\boldsymbol{p})\|^2}{r} \tag{1.2}$$

Following Chebyshev's Inequality,

$$P(\|\tau(\boldsymbol{p}) - \mathcal{G}(\boldsymbol{p})\| \geq \epsilon) \leq \frac{1 - \|\mathcal{G}(\boldsymbol{p})\|^2}{r\epsilon^2} \tag{1.3}$$

Let $\epsilon = f(r)/\sqrt{r}$, where $f(r)$ grows arbitrarily slowly and $\lim_{r \to \infty} f(r) = \infty$

Then, eqn 1.3 becomes

$$\lim_{r \to \infty} P(\|\tau(\boldsymbol{p}) - \mathcal{G}(\boldsymbol{p})\| \geq \epsilon) \leq \lim_{r \to \infty} \frac{1 - \|\mathcal{G}(\boldsymbol{p}\|^2)}{f(r)^2} = 0$$

This suggests distance between $\tau(\boldsymbol{p})$ and $G(\boldsymbol{p})$ might decrease as $1/\sqrt{r}$
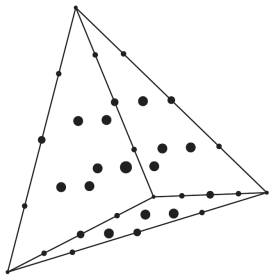
Research Question 1: Distance

Figure 1.2: Population points

Finite population can be only at lattice points
Infinite population can be any where in the space
Distance between finite population and infinite population is bounded by $O(1/\sqrt{r})$

Let η be the random variable $||\tau(\boldsymbol{p}) - G(\boldsymbol{p})||$, and let $\Phi(x) = x^2$
Then from Jensen's Inequality analysis,

$$\mathcal{E}(\|\tau(\boldsymbol{p}) - \mathcal{G}(\boldsymbol{p})\|) \;=\; \mathcal{E}(\eta) \leq \sqrt{\mathcal{E}(\eta^2)} \;=\; \frac{\sqrt{1 - \|\mathcal{G}(\boldsymbol{p})\|^2}}{\sqrt{r}} \qquad (1.7)$$

Distance might decrease as $1/\sqrt{r}$

Diploid Population Model

Diploid genome $\alpha = \langle \alpha_0, \alpha_1 \rangle$

Haploid genome length l

q^n → population at generation n $\qquad$ q^n_α → prevalence of diploid α

$t_\alpha(g)$ → probability that gamete g is produced from parent α

$$q^{n+1}_\gamma = \sum_\alpha q^n_\alpha \, t_\alpha(\gamma_0) \sum_\beta q^n_\beta \, t_\beta(\gamma_1)$$

Diploid Model Reduction to Haploid Model

$$q^{n+1}_{\langle \gamma_0, \gamma_1 \rangle} = p^{n+1}_{\gamma_0} p^{n+1}_{\gamma_1}$$

Where for any haploid γ_0,

$$p^{n+1}_{\gamma_0} = \sum_{\alpha} q^n_{\alpha} t_{\alpha}(\gamma_0)$$

Evolution equation in terms of haploid distributions p,

$$p^{n+1}_{\gamma_0} = \sum_{\alpha_0, \alpha_1} p^n_{\alpha_0} p^n_{\alpha_1} t_{\langle \alpha_0, \alpha_1 \rangle}(\gamma_0)$$

The weighted count of haploid g in generation n is,

$$\sum_{\alpha_0, \alpha_1} q^n_{\langle \alpha_0, \alpha_1 \rangle}([g = \alpha_0] + [g = \alpha_1])$$

$$= \sum_{\alpha_0, \alpha_1} p^n_{\alpha_0} p^n_{\alpha_1}[g = \alpha_0] + \sum_{\alpha_0, \alpha_1} p^n_{\alpha_0} p^n_{\alpha_1}[g = \alpha_1]$$

$$= 2p^n_g$$

Matrix form: $p'_g = p^T M_g p$ where $\left(M_g\right)_{u,v} = t_{\langle u, v \rangle}(g)$

Specialization to Vose's Haploid Model

Mutation: $\boldsymbol{\mu}_i = (\boldsymbol{\mu})^{\mathbf{1}^T i}(1 - \boldsymbol{\mu})^{\ell - \mathbf{1}^T i}$

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

For 1-point crossover,

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

For uniform crossover,

$$c_i = 2^{-\ell}$$

$$t_{\langle u,v \rangle}(g) = \sum_{i \in \mathcal{R}} \sum_{j \in \mathcal{R}} \sum_{k \in \mathcal{R}} \boldsymbol{\mu}_i \boldsymbol{\mu}_j \frac{\chi_k + \chi_{\bar{k}}}{2} [k(u + i) + \bar{k}(v + j) = g]$$

(By Vose & Wright)

$\mathcal{R}$ is set of length *l* binary strings

Walsh Basis

Mixing matrix in Walsh basis

$$\widehat{M}_{u,v} \;=\; 2^{\ell-1}\,[uv = \mathbf{0}]\,\widehat{\boldsymbol{\mu}}_u\widehat{\boldsymbol{\mu}}_v \sum_{k \in \overline{u+v}\mathcal{R}} \boldsymbol{\chi}_{k+u} + \boldsymbol{\chi}_{k+v}$$

Evolution eqn in Walsh basis

$$\widehat{p}_g' \;=\; 2^{\ell/2} \sum_{i \in g\mathcal{R}} \widehat{p}_i\,\widehat{p}_{i+g}\,\widehat{M}_{i,i+g} \qquad \text{where} \quad g\mathcal{R} = \{gi \,|\, i \in \mathcal{R}\}$$

Computational advantages of Reduction to Haploid Model and Walsh Transform

- Computations are simplified due to specialization to Vose's haploid model, which otherwise for diploid case would have been impractical
- Only one mixing matrix as opposed to 2^l is needed to compute next generation
- For $l = 14$, 2^{14} mixing matrices with each having $2^{14} \times 2^{14}$ entries would require 32 terabytes of memory otherwise; whereas one mixing matrix of 2 gigabytes is required

X = 0.1
μ = 0.001
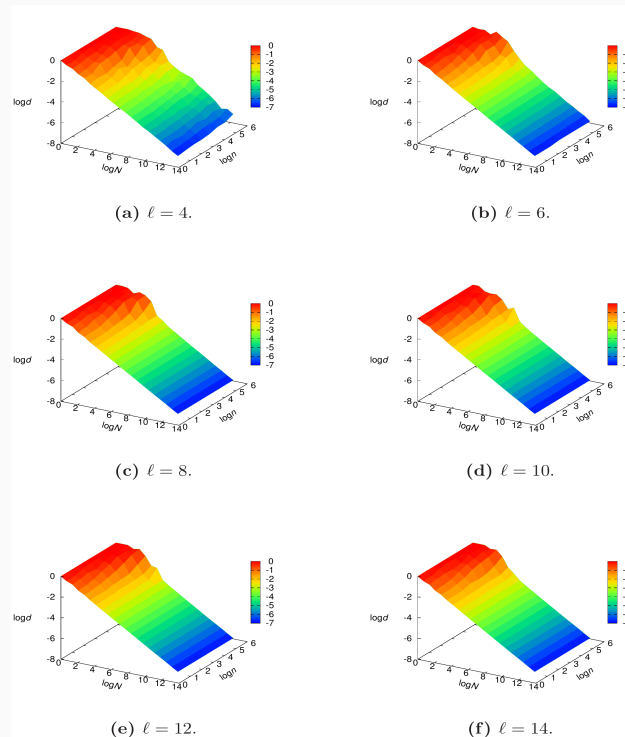
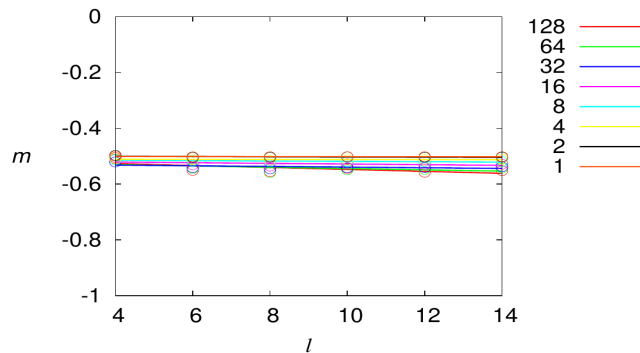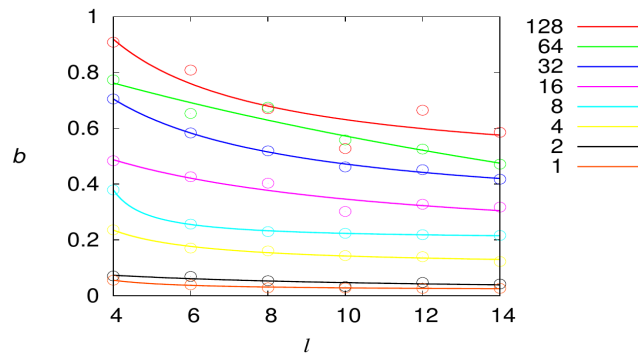Figure: Convergence of finite population behavior

$$\log d = m \log N + b$$

(a) Slope m, genome length ℓ.   **(b)** Intercept b, genome length ℓ.

Figure: Regression parameters for generation n in {1, 2, 4, 8, 16, 32, 64, 128}}

$d \approx N^m e^b$
From figure (a) above, m = -0.5
$d \approx k / \sqrt{N}$

- Vose's infinite population model can be reduced to diploid case
- Distance between finite diploid population and infinite diploid population decrease like $1/\sqrt{N}$

Research Question 2:

Oscillation in Finite Population Evolution

If G is continuously differentiable,
the sequence $\boldsymbol{p}$, $G(\boldsymbol{p})$, $G^2(\boldsymbol{p})$, … converges to a fixed point (also called limit)

$$\mathcal{G}(\omega) = \lim_{n \to \infty} \mathcal{G}^n(\boldsymbol{p}) = \omega$$

But under some circumstances, the sequence converges to periodic
orbits that oscillate between two fixed points ($\boldsymbol{p}*$ & $\boldsymbol{q}*$)

For any $g \in \mathcal{R}$, $g \neq 0$

$$-1 = \sum_j (-1)^{g^T j} \boldsymbol{\mu}_j$$

$$1 = \sum_{k \in \bar{g}\mathcal{R}} \boldsymbol{\chi}_{k+g} + \boldsymbol{\chi}_k$$

Does Finite Population Evolution Also Oscillate?

If finite population follows infinite population behavior,
does it also show oscillation?

Simulations run for both haploid and diploid population simulation
Genome lengths $l \in \{8, 10, 12, 14\}$ were used
Random initial population
Finite population size 4096, 40960, 81920 used
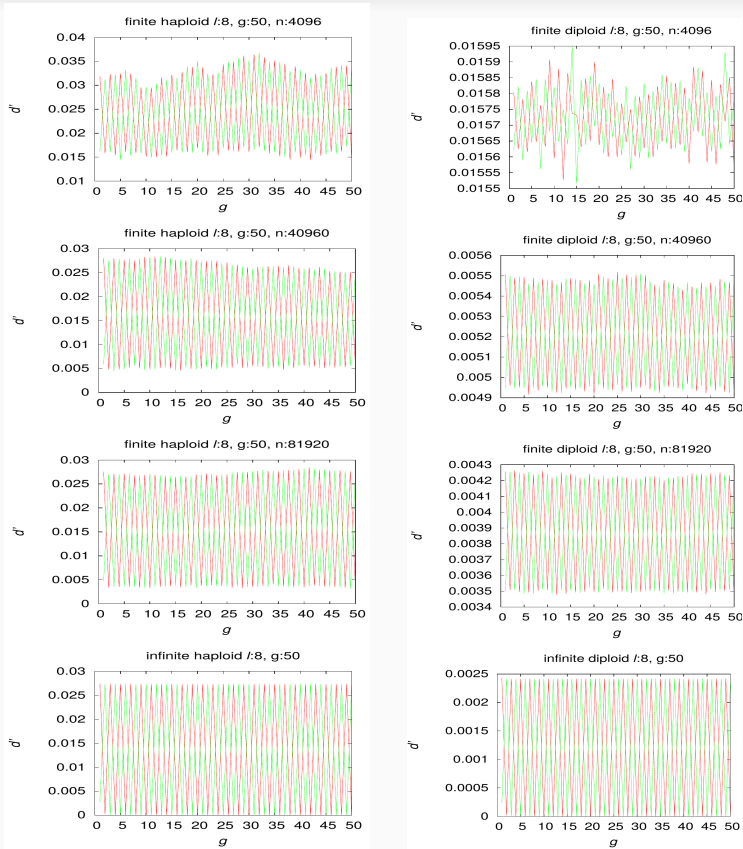Distance between fixed points and population used to visualize oscillation

Figure: Oscillation for *l = 8*
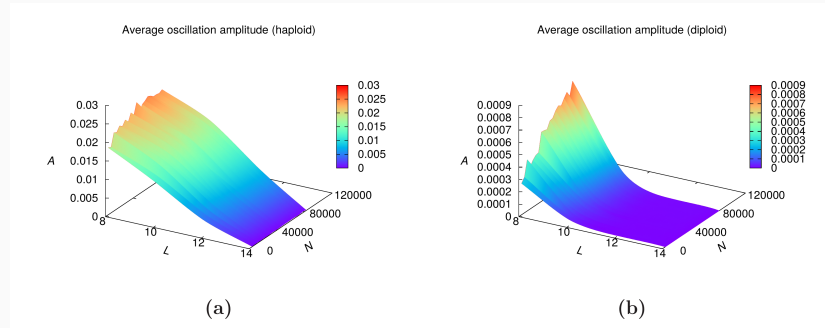
Figure: Oscillation for *l = 12*

Figure: Average oscillation amplitude

As population size increases, oscillation approaches the behavior exhibited by infinite population
Oscillation amplitude increases with increase in population size
Amplitude of oscillation decreases with increase in *l* and so does quality of oscillation