Efficient Simulation Of A Simple Evolutionary System

Mahendra Duwal Shrestha

The University Of Tenessee

February 28, 2017

Outline

Background

Question 1: Distance between finite and infinite population

Question 2: Oscillation in finite population

Question 3: Oscillation in finite population under violation in mutation

Question 4: Oscillation in finite population under violation in crossover

Conclusion

Terms

Population P: a collection of length ℓ binary strings

Population vector $\mathbf{p}$: $\mathbf{p}_j$ is the proportion of string j in the population

If $P = 00, 01, 01, 10, 11, 11$, then $\mathbf{p}_3 = 2/6 = 1/3$

$\mathcal{R}$ denotes a set binary strings of length ℓ

- Addition and multiplication of elements in $\mathcal{R}$ are bitwise operations modulo 2
- $x = 1101$, $y = 1010$
- $x + y = 1101 + 1010 = 0111$
- $xy = 1101 \cdot 1010 = 1000$
- $\bar{x} = 0010$

Crossover & Mutation

Crossover : Choose parents u and v, exchange bits using crossover mask m:

$u' = um + v\bar{m}, v' = u\bar{m} + vm$

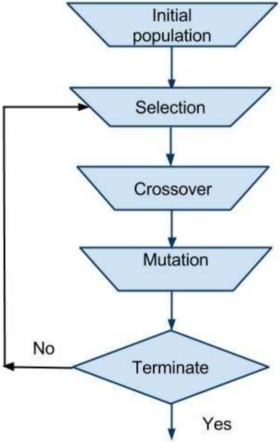
$u = \mathbf{11001011}, v = 11011111, m = 11110000$

$\{\mathbf{11001011}, 11011111\} \to \{\mathbf{1100}0000 + 00001111,\ 0000\mathbf{1011} + 11010000\} \to \{\mathbf{1100}1111, 1101\mathbf{1011}\}$

Mutation: Flip bits using mutation mask:

$x \to x + m$

Finite Population GA



Randomly select parents u and v

Crossover u and v to produce u' and v'

Keep one of u', v' , and mutate

Repeat above to form next generation

Repeat whole process until system stops to improve or threshold is reached

Infinite Population Model

Population is modeled as a vector $\mathbf{p}$

$\mathcal{G}$ maps $\mathbf{p}$ to the next generation

$\mathcal{G}(\mathbf{p})_j =$ probability that string j occurs in the next generation

The infinite population model is the sequence

$\mathbf{p} \to \mathcal{G}(\mathbf{p}) \to \mathcal{G}(\mathcal{G}(\mathbf{p})) \to \cdots$