CSE/ECE 848 Introduction to Evolutionary Computation

Module 3 - Lecture 15 - Part 1

Expectation Distribution Algorithms (EDA)

Wolfgang Banzhaf, CSE
John R. Koza Chair in Genetic Programming

EDAs

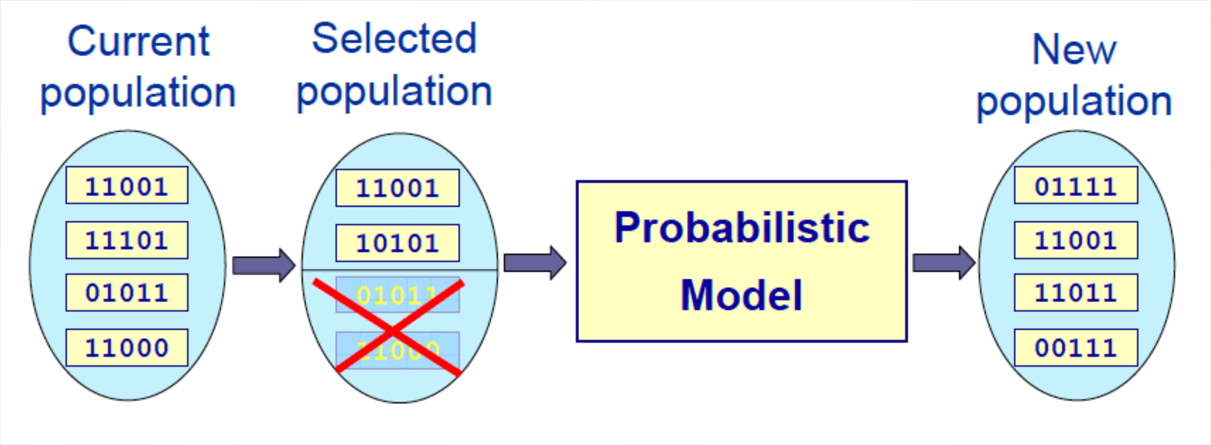
- Also known as:
  - Probabilistic Model Building Algorithms
  - Iterated Density Estimation Algorithms
- Given a population of candidate solutions with their fitness, what solutions to generate next?
- GA example

#	Solution	Evaluation
1	00100	1
2	11011	4
3	01101	0
4	10111	3

→ ??

PMBGA

- Replace mutation and crossover with learning and sampling a probabilistic model



History

- EDAs: Muehlenbein and Paass, 1996
- IDEA: Bosman and Thierens, 2000
- PMBGA: Pelikan and Goldberg, 1998 (BOA)

Simplest Models

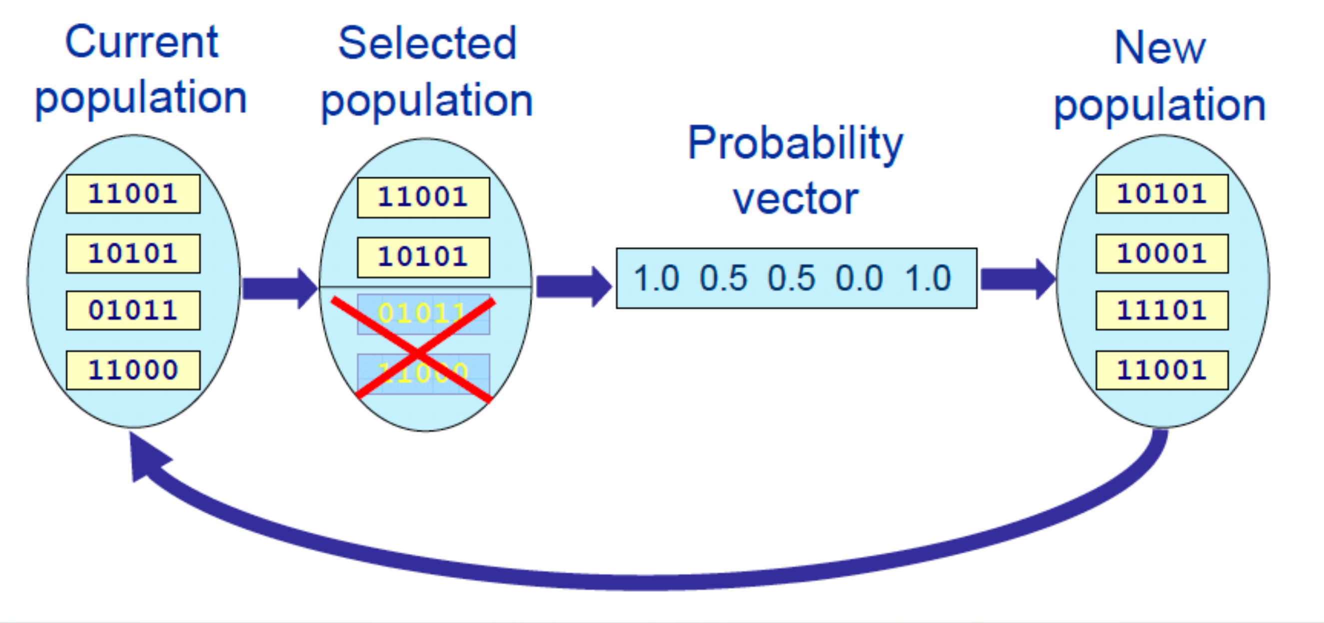
- Assume a n-dim bitstring for a solution (classical GA type)
- Model: Probability string or vector of n dimensions:

  $p=(p_1, p_2, \dots p_n)$

  where each p_i means the probability of 1 in dimension i
- Learn p by computing proportion of 1 in positions 1 … n
- Produce new string population by sampling from p

Simplest Models II

- Iterate model sampling and building process



Features

Advantages

- Bits that perform better receive more copies in the next generation
- They are combined in new ways, potentially leading to the global optimum

Disadvantages

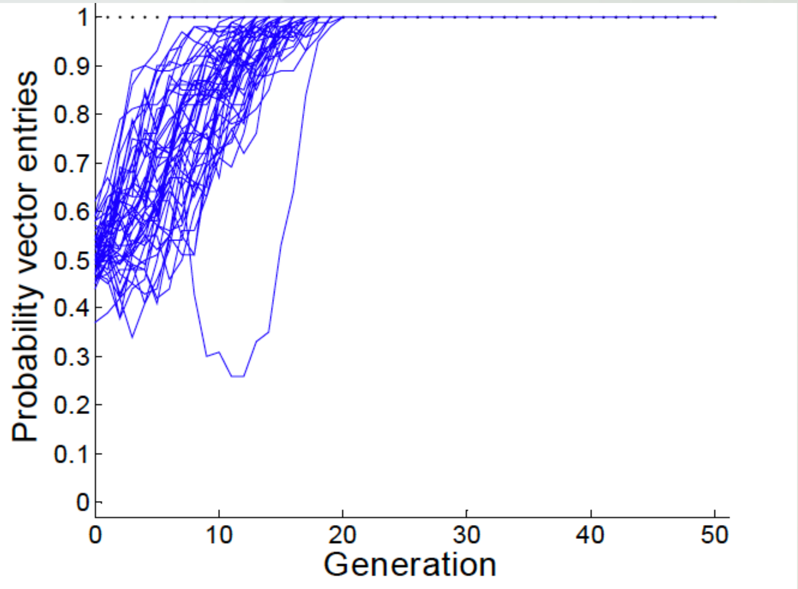
- But: Context of each bit is ignored! So works only for special problems without correlations between components

Example

- One-Max Problem

$$f(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} X_i$$

- Probability vector on One-Max

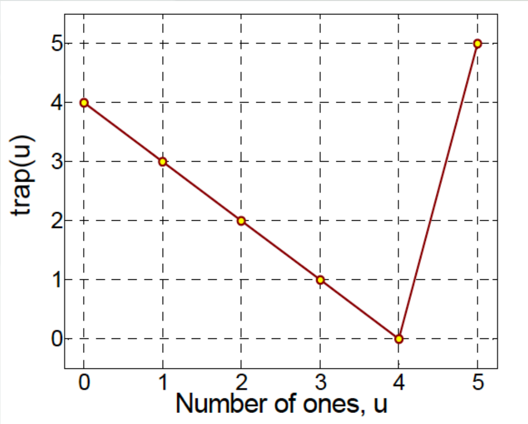


- Ideal scale up of N log (N) evaluations required to converge

A Deceptive Problem

- Form a trap function

$$trap(ones) = \begin{cases} 5 & \text{if } ones = 5 \\ 4 - ones & \text{otherwise} \end{cases}$$

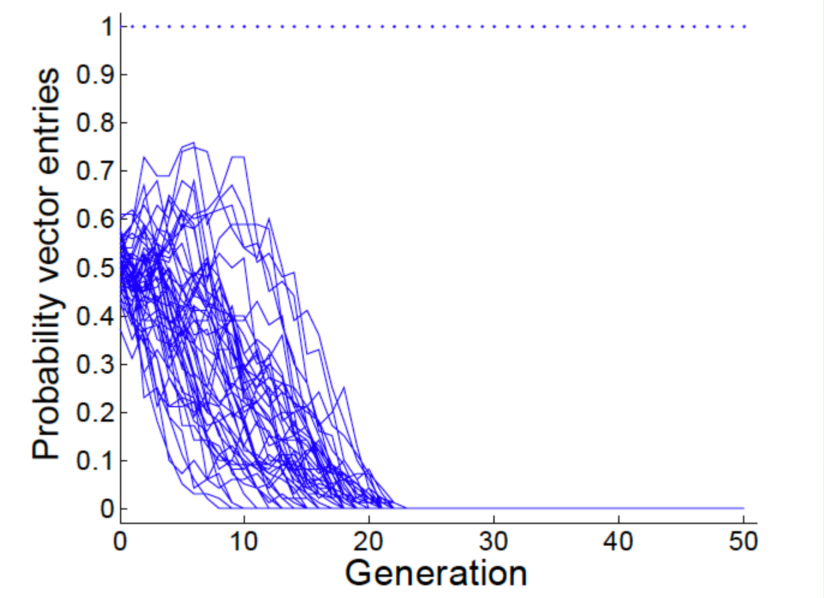


Trap-5

- Concatenated traps:
  - Partition string into disjoint groups of 5 bits which contribute to the fitness function
  - Sum all 5 bit traps to provide overall fitness
  - Optimum string: (11111,…..,11111)

A Deceptive Problem II

- Single bits are misleading!
- f(0****)=2 vs. f(1****)=1.375



A Remedy

- Consider 5-bit statistics!
- Thus 11111 will outperform 00000
- Learn model p(00000), p(00001), …, p(11111)
- Sample 5 bits at a time, ie

  generate 00000 with p(00000), 00001 with p(00001) etc.
- However, we would need to learn the relevant context!