GNR602 Advanced Methods in Satellite Image Processing

Instructor: Prof. B. Krishna Mohan

CSRE, IIT Bombay

bkmohan@csre.iitb.ac.in

Slot 13

Lecture 21 Introduction to Genetic Algorithms

Contents of the Lecture

- Introduction to Genetic Algorithms
- Genetic Operators
- Cost Functions
- Applications

Genetic Algorithms

- Genetic algorithms are one of the well known tools for optimization.
- They are employed to generate optimal solutions employing the principles of genetic evolution.
- They employ the concept of random search instead of deterministic search

Genetic Algorithms

- Genetic algorithms are inspired by evolutionary processes in living organisms.
- 27 videos of a full course on this subject taught by Prof. David Delchamps (Cornell University) can be found on Youtube.
- Search key on Youtube: **ECE4271 David Delchamps Evolutionary Processes**

Basic Principle

- Numerical approach to problem solving
- Uses genetics as its model of problem solving.
- Apply rules of reproduction, gene crossover, and mutation
- Start with a number of candidate solutions that evolve over “genetic cycles” towards the optimal solution
- Solutions evaluated using “fitness” criteria
- Fittest will survive

Genetic Algorithms

Different from traditional methods

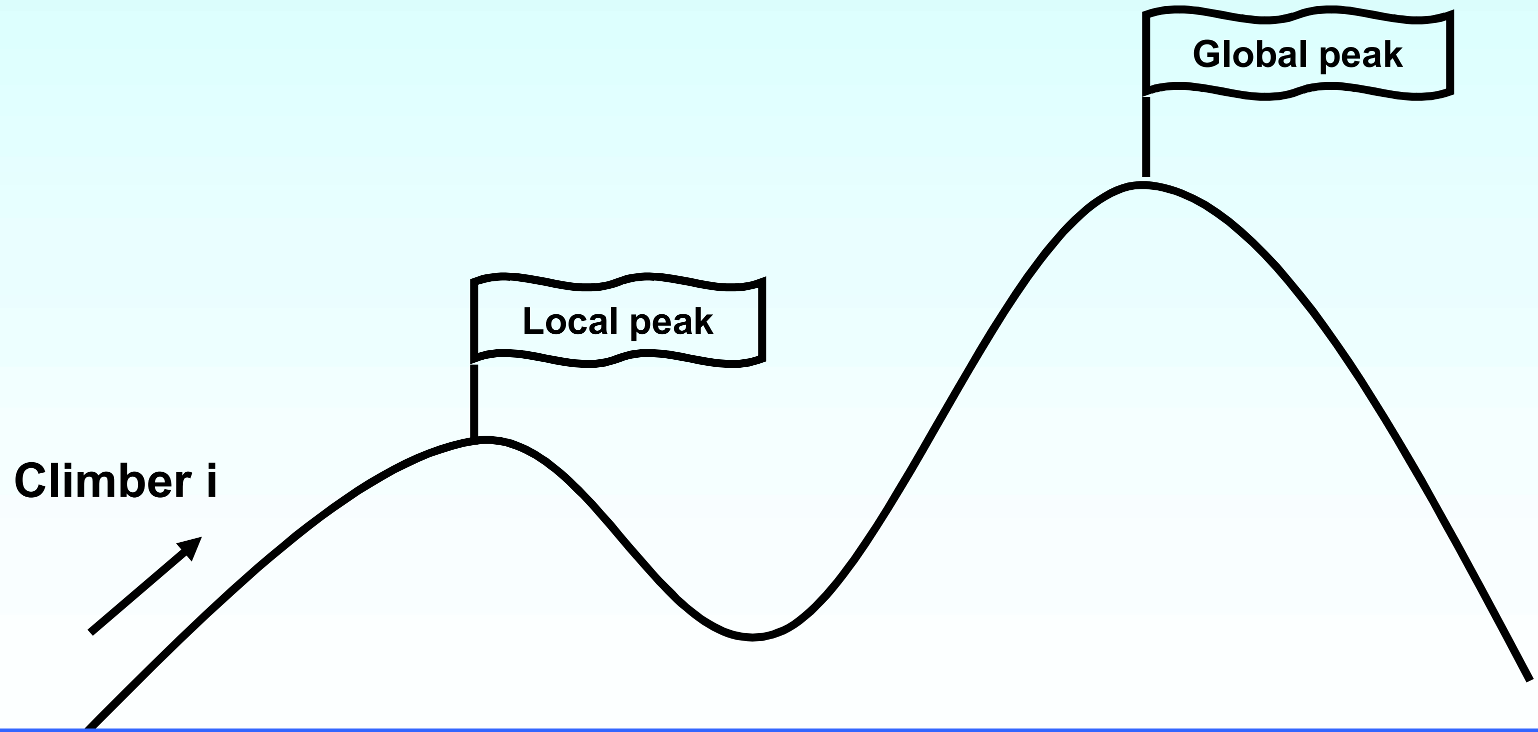
- Work with a coding of the parameter set
- Search from a population of points
- Use payoff (objective function) information
- Use probabilistic transition rules

Genetic Algorithm

- A computer algorithm inspired by the principles of genetic evolution

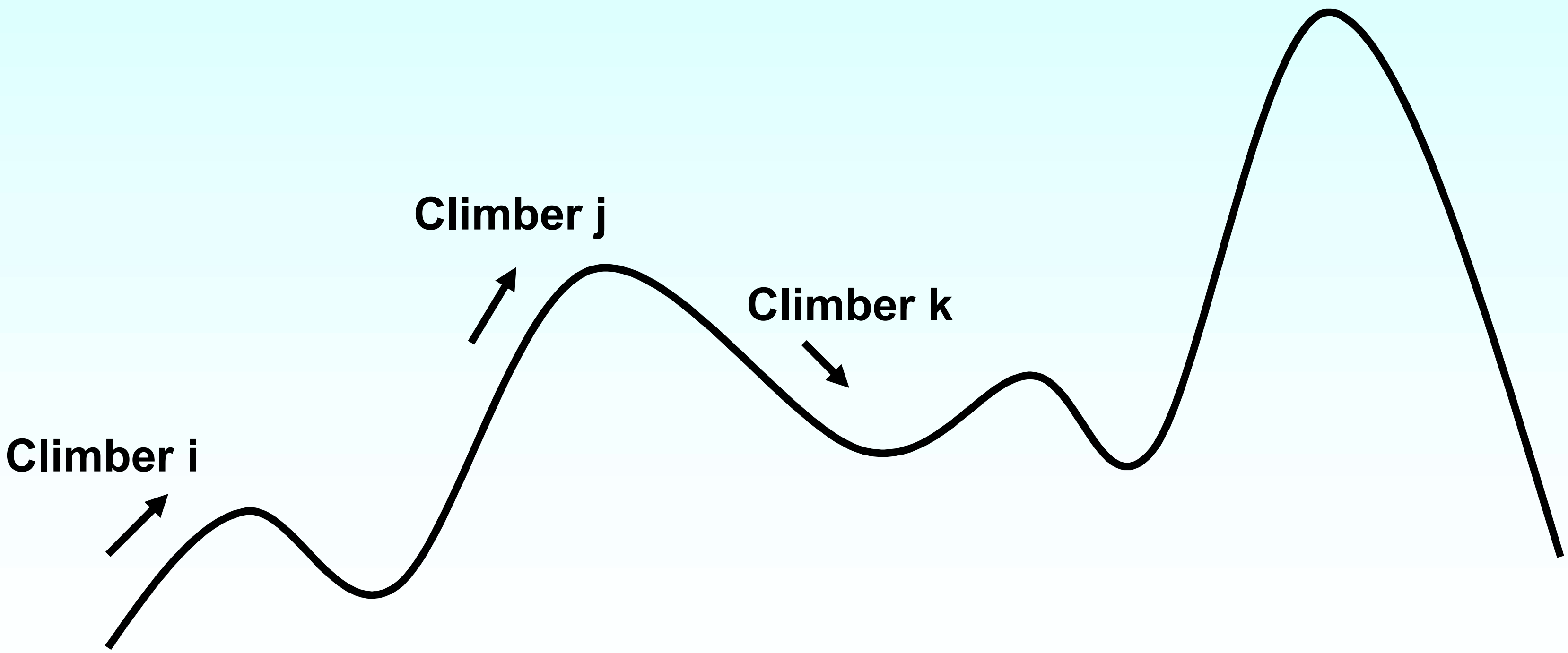
Genetic Algorithm Approach

- Maximization by Hill climbing



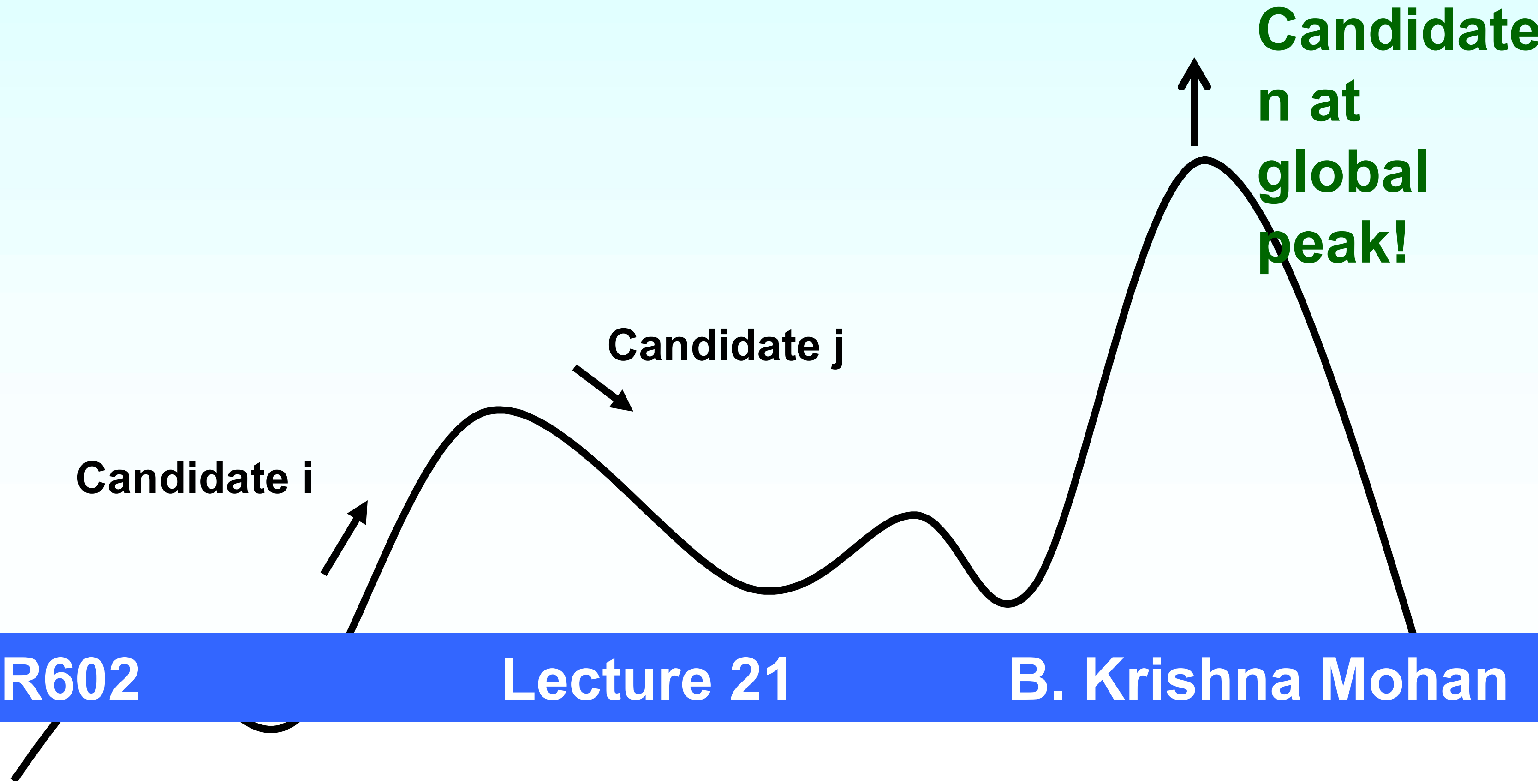
Multiple Candidates

- Multi-climbers



Motivation

- In course of time at least one candidate may reach the global peak



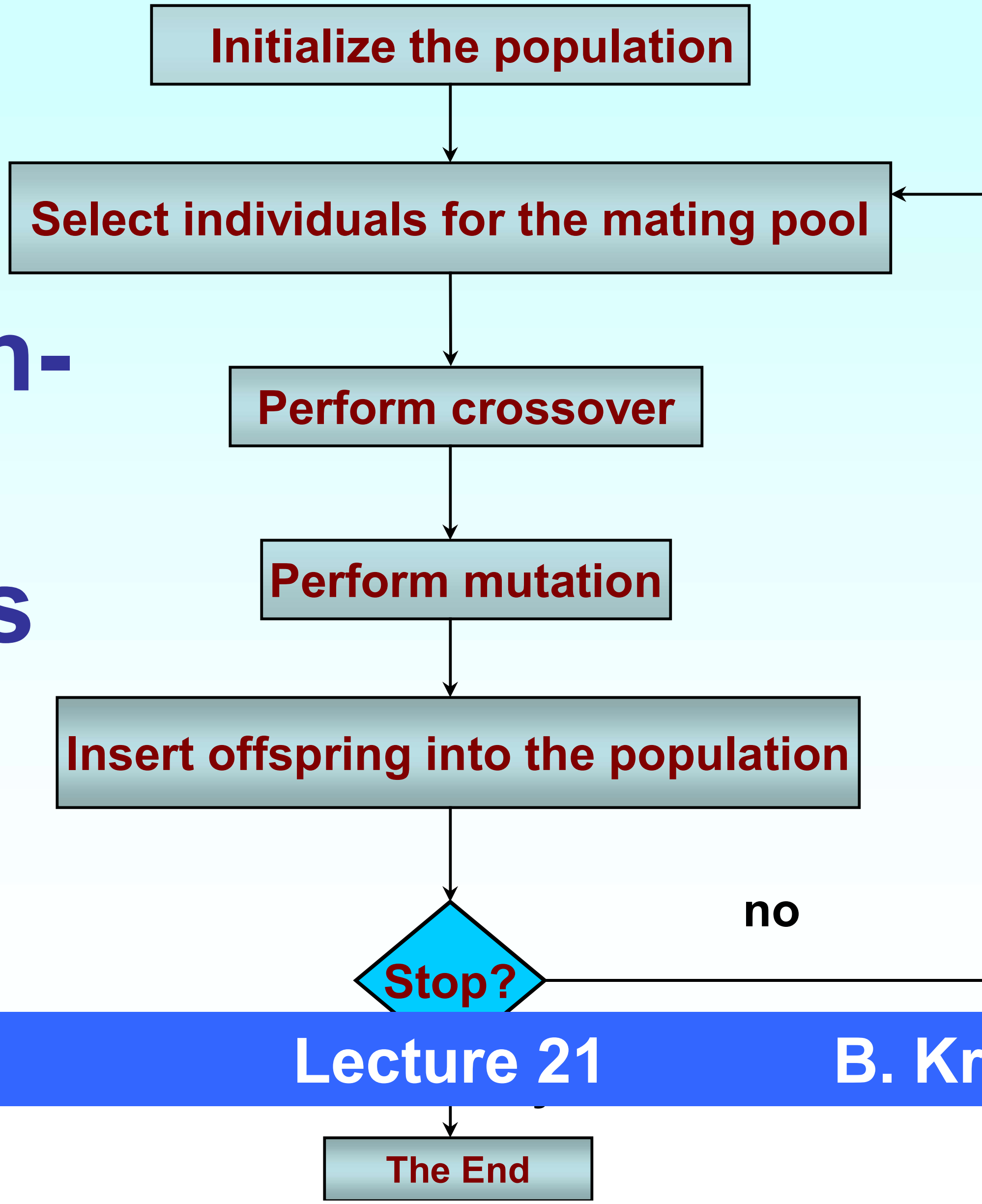
GA based Search

- Based on the principle of evolution and genetics into search .
- Simulate the process in natural systems.
- Create a population of individuals represented by chromosomes, in essence a set of character strings, that are analogous to the DNA, that we have in our own chromosomes.

Survival of the Fittest

- The main principle of evolution used in GA is ***"survival of the fittest".***
- The good solutions survive, while bad ones die.
- The definition of ***fitness*** is application dependent

Algorith-mic Phases



Designing GA...

- How to represent genomes?
- How to define the crossover operator?
- How to define the mutation operator?
- How to define fitness function?
- How to generate next generation?
- How to define stopping criteria?

Crossover

- Crossover is concept from genetics.
- Crossover combines genetic material from two parents, in order to produce superior offspring.
- A few types of crossover:
  - One-point
  - Multiple point.

One-point Crossover

One-point Crossover

Child #1	Child #2
0	7
1	6
5	2
3	4
4	3
5	2
6	1
7	0

Mutation

- Introduces randomness into population.
- Unary operation.
- The idea is to reintroduce divergence into a converging population.
- Performed on small part of population, in order to avoid entering unstable state.

Mutation

Parent	1	1	0	1	0	0	0	1
Child	0	1	0	1	0	1	0	1

Probabilistic Operations

- Average probability for individual to crossover is about 80%.
- Average probability for individual to mutate is about 1-2%.
- Probability of genetic operators follow the probability in natural systems.
- Better solutions reproduce more often.

Fitness Function

- Fitness function is evaluation function, that determines what solutions are better than others.
- Fitness is computed for each individual.
- Fitness function is application dependent.

Selection

- Copies a single individual, probabilistically selected based on fitness, into the next generation
- Possible ways of selection:
  - "Only the strongest survive"
  - "Some weak solutions survive"
    - Assign a probability that a particular individual will be selected for the next generation
    - More diversity
    - Some bad solutions might have good parts!

Stopping Criteria

Two possible solutions:

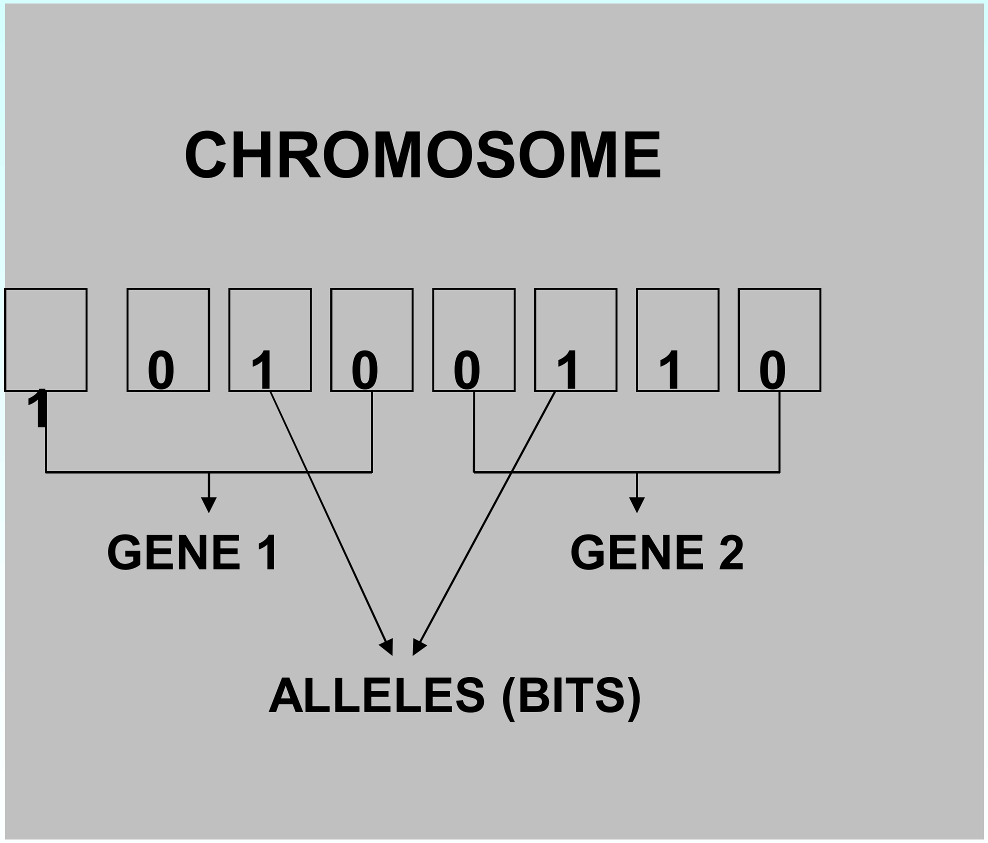
- Stop after production of definite number of generations
- Stop when the improvement in average fitness over two generations is below a threshold

Some Details

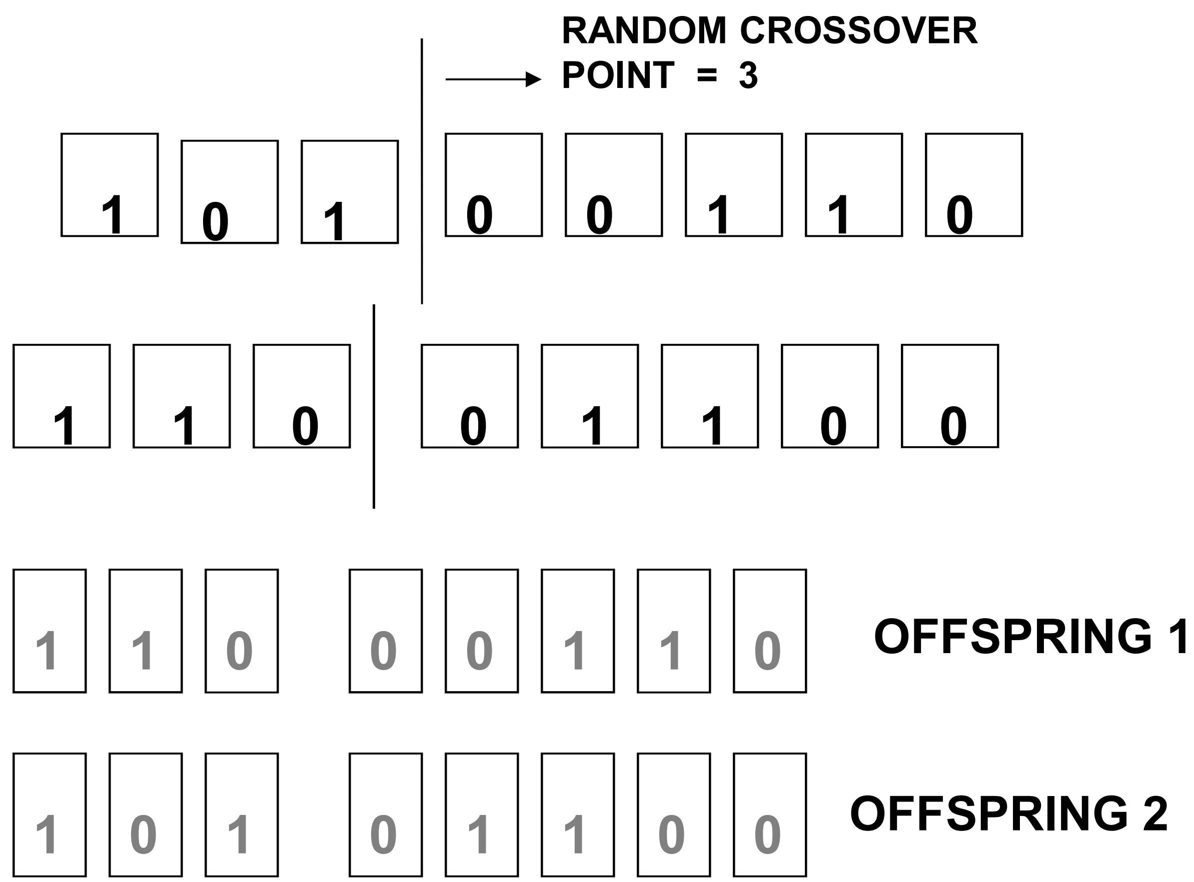
Advantages of GAs

- Concept is easy to understand.
- Minimum human involvement.
- Find new solution!
- Modular, separate from application
- Supports multi-objective optimization
- Easy to exploit previous or alternate solutions

Data Structure For GA



Simple Crossover Step



Multi-point Crossover

Chromosome I		Chromosome II
123 |456 |78	Bit Position	123 |456 | 78
110 |001 |01	Bit Value	100 |001 | 10

Randomly selected two points for crossover

123 |456 |78	Bit Position	123 |456 |78
110 |001 |10	Bit Value	100 |001 |01
Offspring I		Offspring II

Shuffling Crossover

CHROMOSOME I		CHROMOSOME II	
12345678 10010011	Bit Position Bit Value	12345678 10000110	
15764832 10101100		27465318 01010010	Random Shuffling
157 | 64832 101 | 01100		274 | 65318 010 | 10010	Random Crossover Site
15764832 10110010		27465318 01010010	Generation of Offsprings
12345678 10100110		12345678 10000110	Reordering of bit positions to generate the final

Single Point Mutation

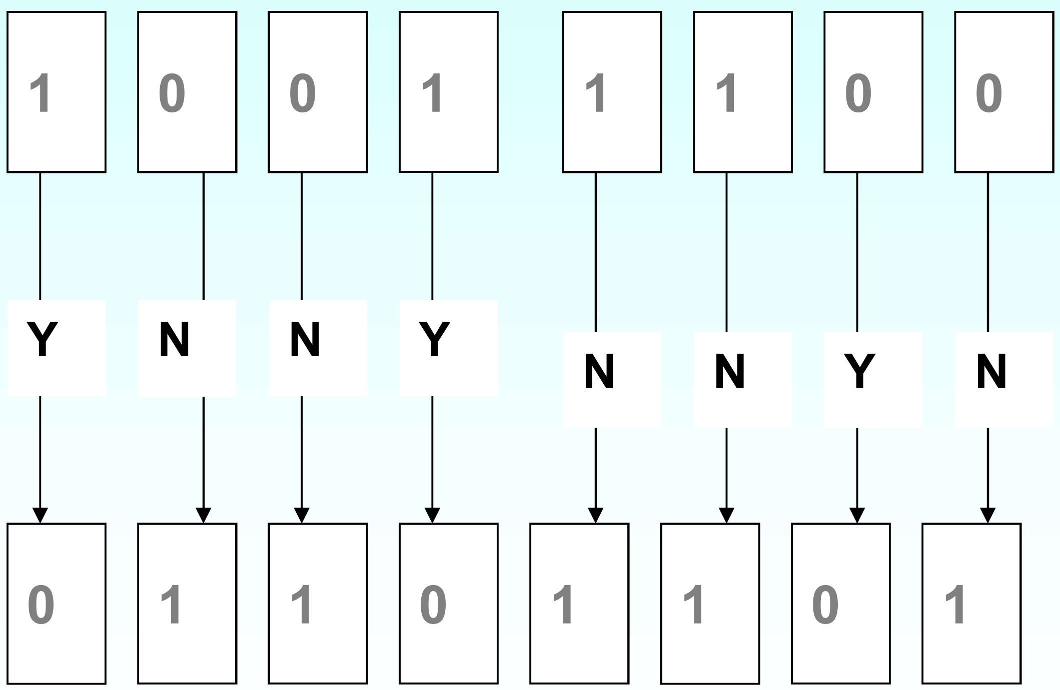
1	0	0	1	1	1	0	0

RANDOM MUTATION POINT

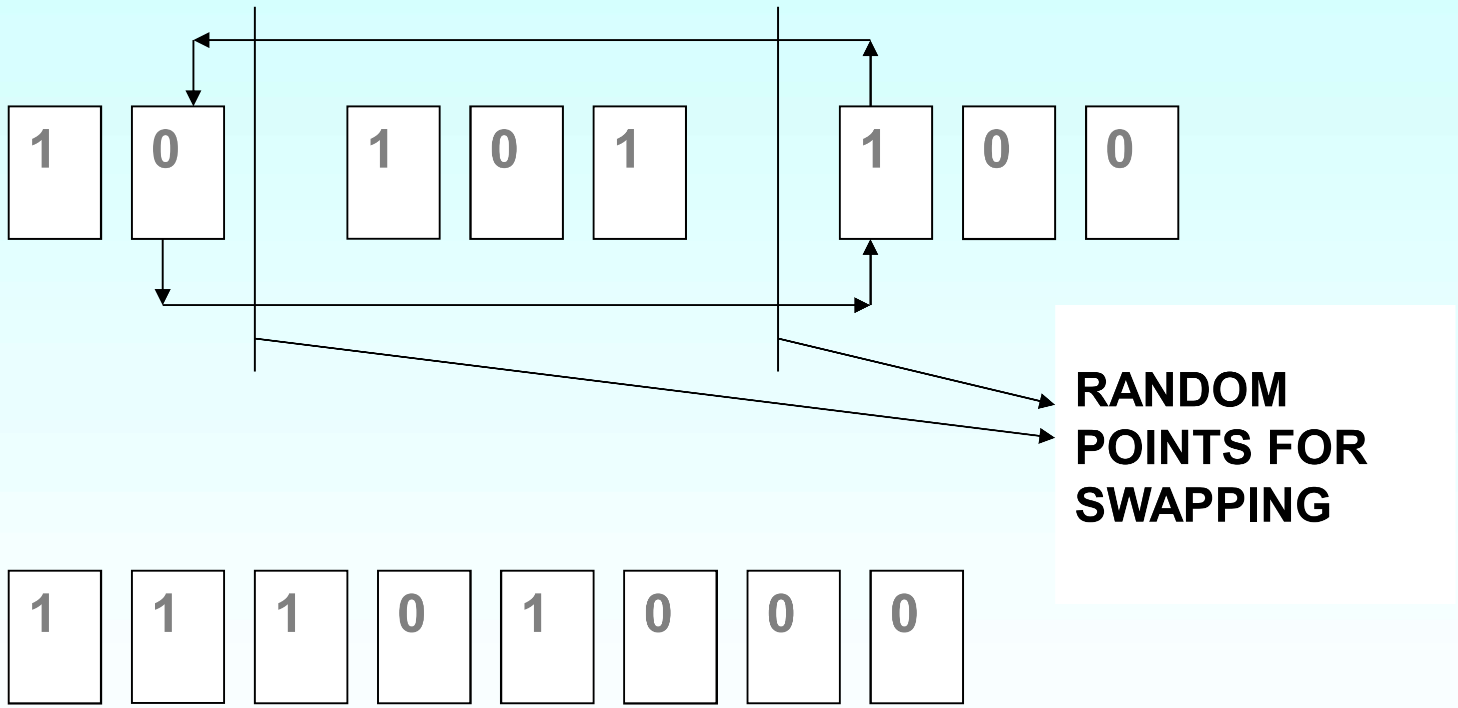
1	0	0	0	1	1	0	0

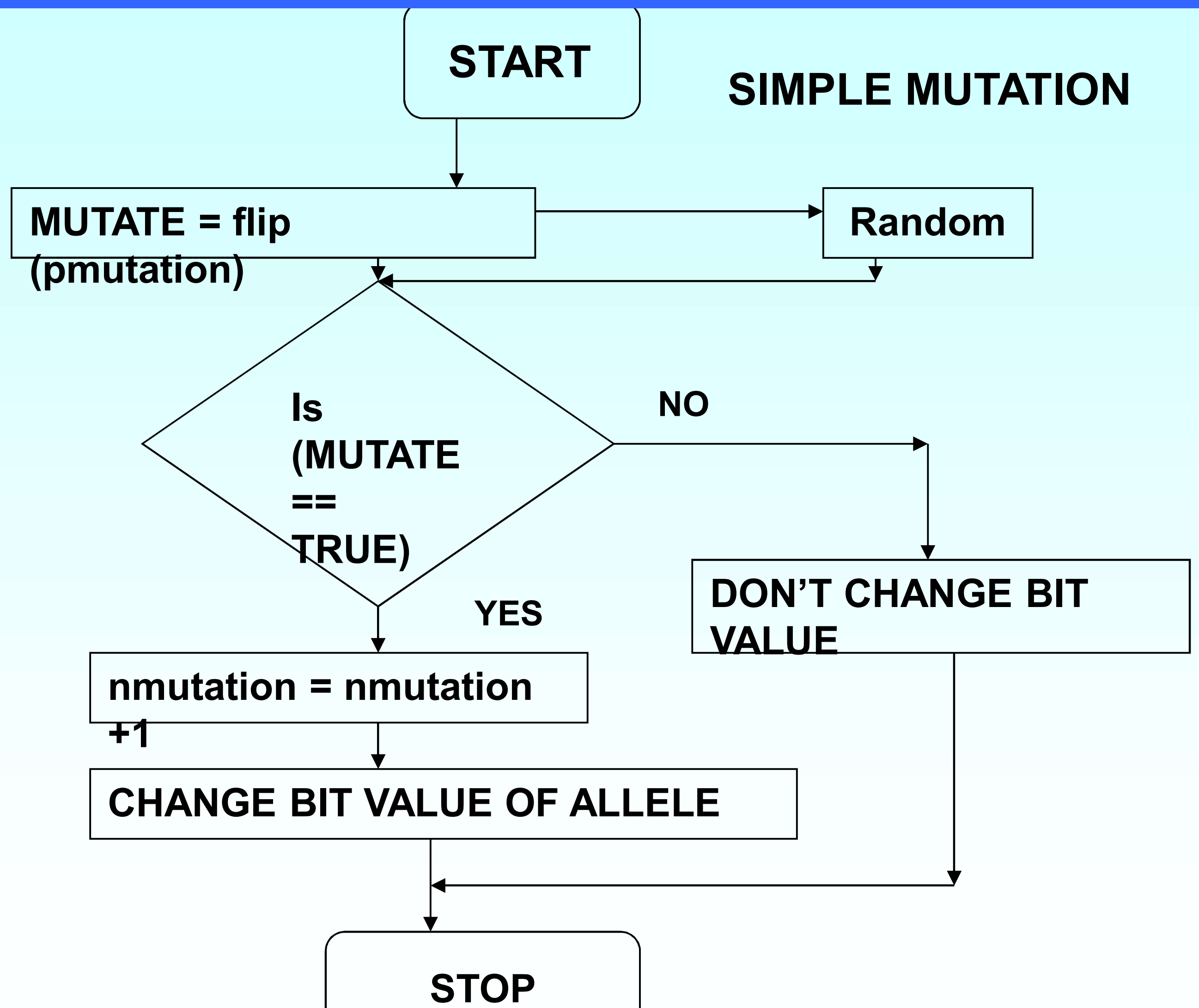
AFTER MUTATION

Uniform Mutation



Swap Mutation




START
SIMPLE MUTATION
MUTATE = flip (pmutation)
Random
Is (MUTATE == TRUE)
NO
YES
DON'T CHANGE BIT VALUE
nmutation = nmutation +1
CHANGE BIT VALUE OF ALLELE
STOP

Fitness Function

- Cost associated with a weight set =

  Average error in classification for the entire set of test samples
- Lower error = Higher fitness
- Using a number of candidate weight sets, a multilayer perceptron network is initialized.

Image classification application using ANN

Cost Function for Rate Controlled JPEG

$$f(x) = \alpha * (MSE) + (1 - \alpha) * (size.error)$$

α is the weight varying from 0.0 to 1.0

Size error =abs((compressed file size) - (desired file size))

$$MSE = \sum_{i=0}^{i=nl*np} (x_i - \hat{x}_i)^2 /(nl * np)$$

Fitness Function = -Cost Function

Fitness Function

- Fitness associated with a band subset =

  Accuracy of classification for the entire set of test samples + Separability of classes

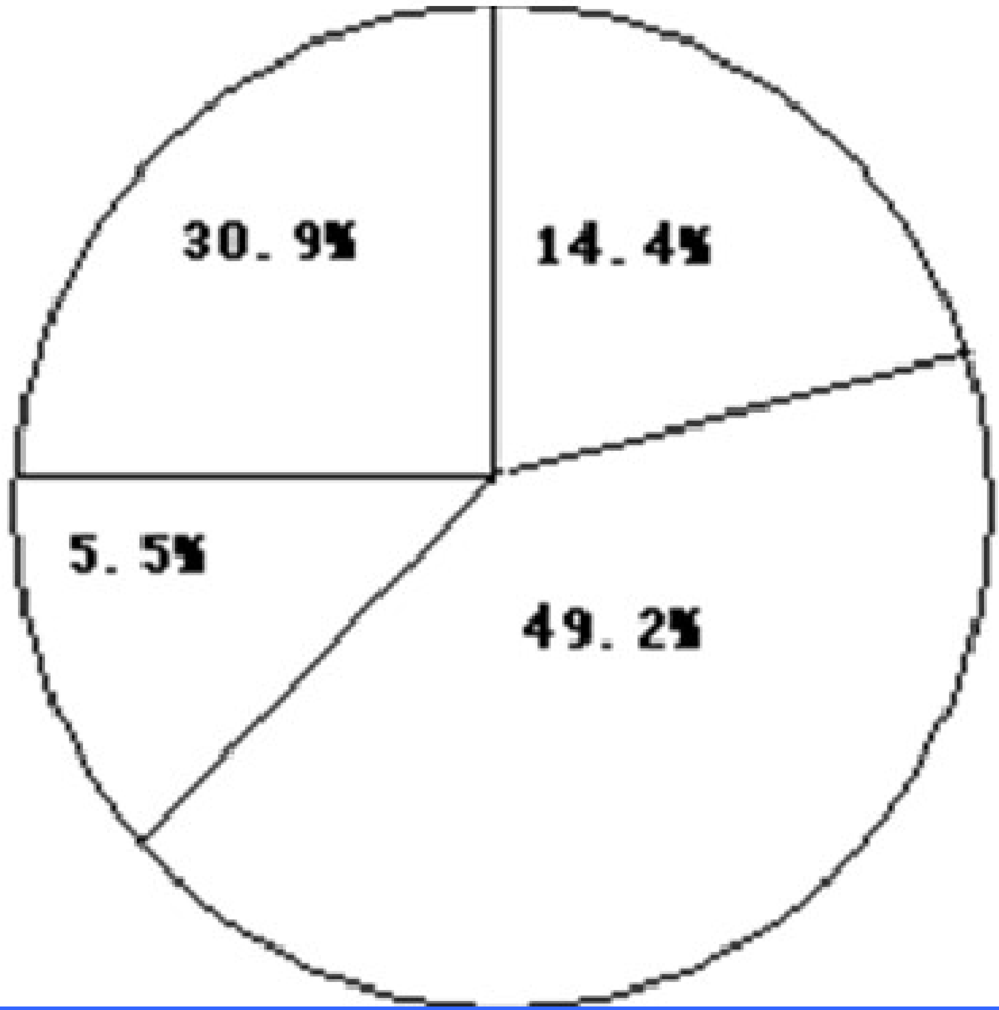
- High accuracy + High separability = High fitness

Fitness for Dimensionality Reduction

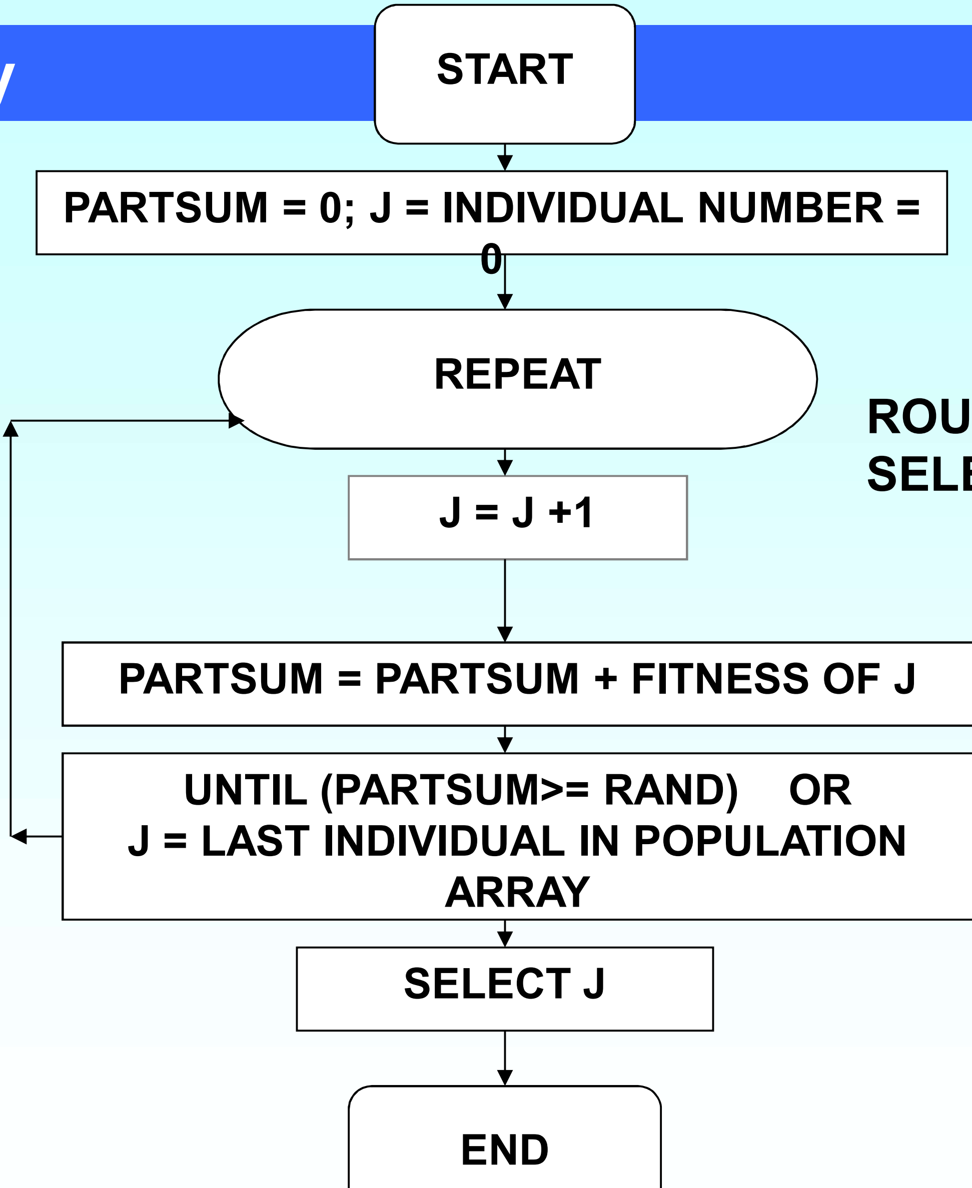
Selection

- Roulette wheel selection
- Rank selection
- Tournament selection

Roulette Wheel Selection



ROULETTE WHEEL SELECTION



Rank Selection

START

↓

COMBINE ALL 2*n* MEMBERS FROM BOTH OLD AND NEW POPULATION AND RANK THEM ACCORDING TO THEIR FITNESSES.

↓

SELECT FIRST *n* RANKS AS THE POPULATION FOR THE NEXT ITERATION

↓

END

Tournament Selection

START

↓

COMBINE ALL *2n* MEMBERS OF BOTH OLD AND NEW POPULATION

↓

FROM SUCH *2n* MEMBERS RANDOMLY SELECT 2 MEMBERS

↓

Compare the fitnesses between members and select the fitter one while return the other member back to the population

↓

CONTINUE THE PROCEDURE UNTIL *n* MEMBERS ARE SELECTED FOR THE NEW ITERATION

↓

STOP

Convergence Criteria

Image Classification

- Accept the candidate if the corresponding error in classification is within the user specified tolerance limit (e.g. accuracy > 95% or error < 5%)

Rate control in JPEG Method

- When is a precise compression rate essential?
  - Maximum quality to be preserved
  - Full capacity of communication channel to be utilized
  - Data rate should not exceed the channel capacity

Key requirement in space applications

How to control rate of compression with JPEG?

- Quality factor is the only way to compress less or more with JPEG.
- How can the desired rate of compression be specified in JPEG?
- What is the implication when the rate of compression is user specified on JPEG technique?

Quality Factor in JPEG

- The information loss happens during the quantization phase
- In normal case quantization table coefficients are scaled up or down based on user-specified
- The precise rate cannot be predicted based on the quality factor

Role of GA in JPEG

- Predict the quantization table given a compression factor
- Fitness function based on –
  – Quality of image after decompression
  – Size of the compressed image
  – Variable weightage to the two factors based on user requirement

Approach

- Generate a large number of candidate quantization tables
- Encode quantization tables
- Apply Crossover
- Apply Mutation
- Compress image
- Check size of compressed image

Approach contd.

- Decompress the compressed image
- Compare decompressed image with original
- Note distortion in pixel values
- Fitness based on
  - Minimum deviation of compressed file size
  - Minimum distortion due to compression

Convergence Criteria

Image Compression

- Accept the member if the corresponding error in file size is within the user specified tolerance limit (e.g., 80 bytes for image size upto 1MB and 5000 bytes for image size > 20 MB)

Some results

Sample Input Data

- InputDataFileForGA01
- InputNodes 4; HiddenLayers 2
- HiddenNodes 16 13; OutPutNodes 7
- PopulationSize 20; No.OfGenerations 200
- SearchMinValue -5.0; SearchMaxValue +5.0
- AllowbleError 0.01; CrossOverProbability 0.80
- MutationProbability 0.1
- TrainingDataFileName rajtrpat.dat
- NetWorkWeightsFile raj.wgt

Sample Input Data

- InputDataFileForGA01
- InputNodes 4; HiddenLayers 2
- HiddenNodes 16 19; OutPutNodes 15
- PopulationSize 20; No.OfGenerations 20
- SearchMinValue -2.0; SearchMaxValue +2.0
- AllowbleError 0.01; CrossOverProbability 0.80
- MutationProbability 0.1
- TrainingDataFileName kdatrpat.dat
- NetWorkWeightsFile kdaga.wgt

What is the candidate for Neural network classification?

- A candidate is the collection of all network link weights. For example, consider that there are
  - n1 input nodes,
  - n2 nodes in 1st hidden layer,
  - n3 nodes in 2nd hidden layer and
  - n4 nodes in the output layer
- A candidate in this will be a collection of n real numbers, where n = n1xn2 + n2xn3 + n3xn4
- In binary form, a candidate will be nx32 or nx64 bits according to single or double precision real numbers

What is the candidate for Dimensionality Reduction?

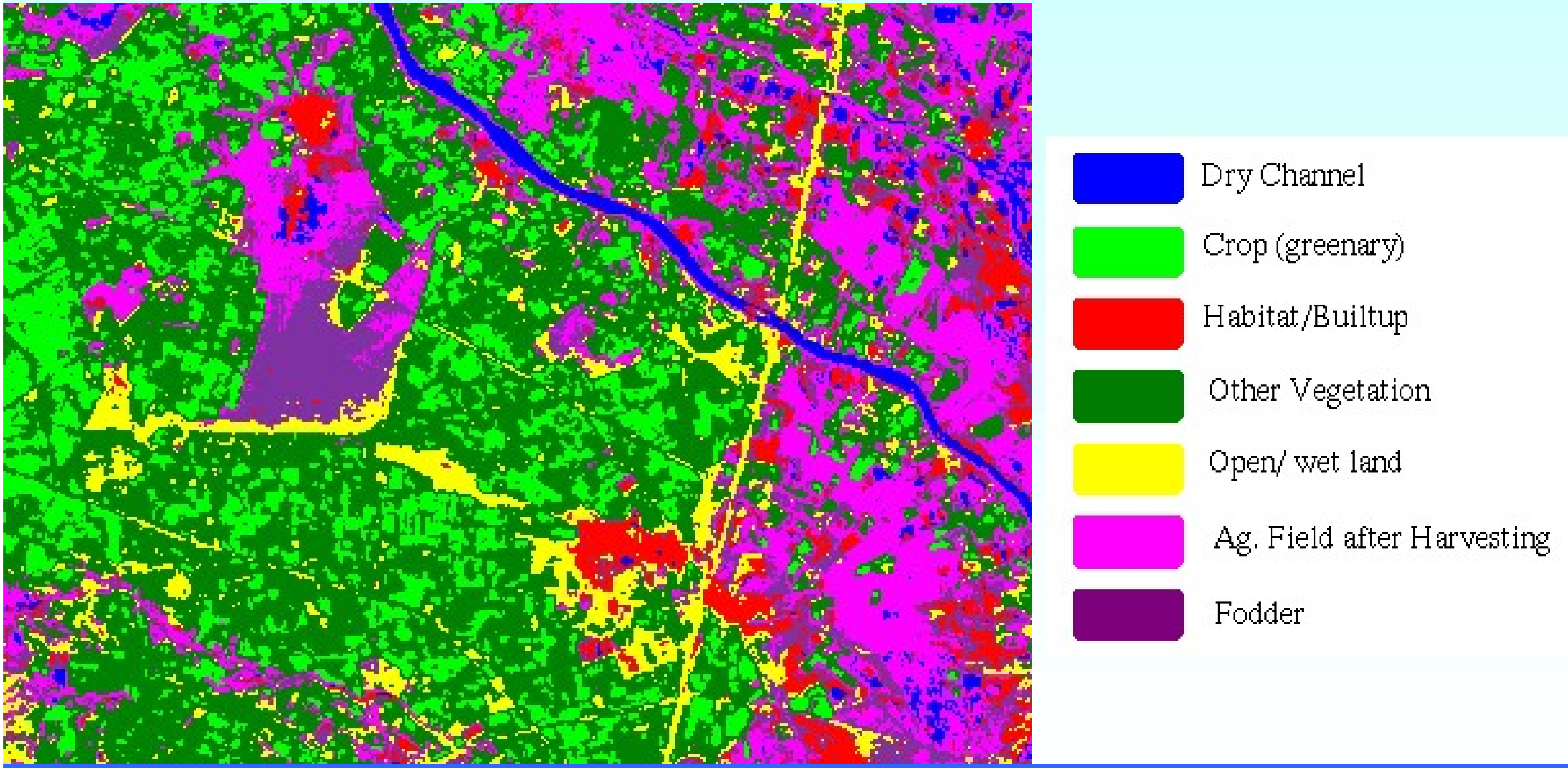
- A candidate is the collection of n integers, where each integer is the serial number of a particular band in the hyperspectral data set.
  - N bands
  - L desired bands
- A candidate here will be a sequence of integers, given by $n_1, n_2, \ldots, n_L$ where any n_k lies in the range [1 N].
- In integer representation a candidate is a binary string of Lx8 bits assuming maximum number of bands N does not exceed $2^8 = 256$.

What is the candidate for JPEG Compression?

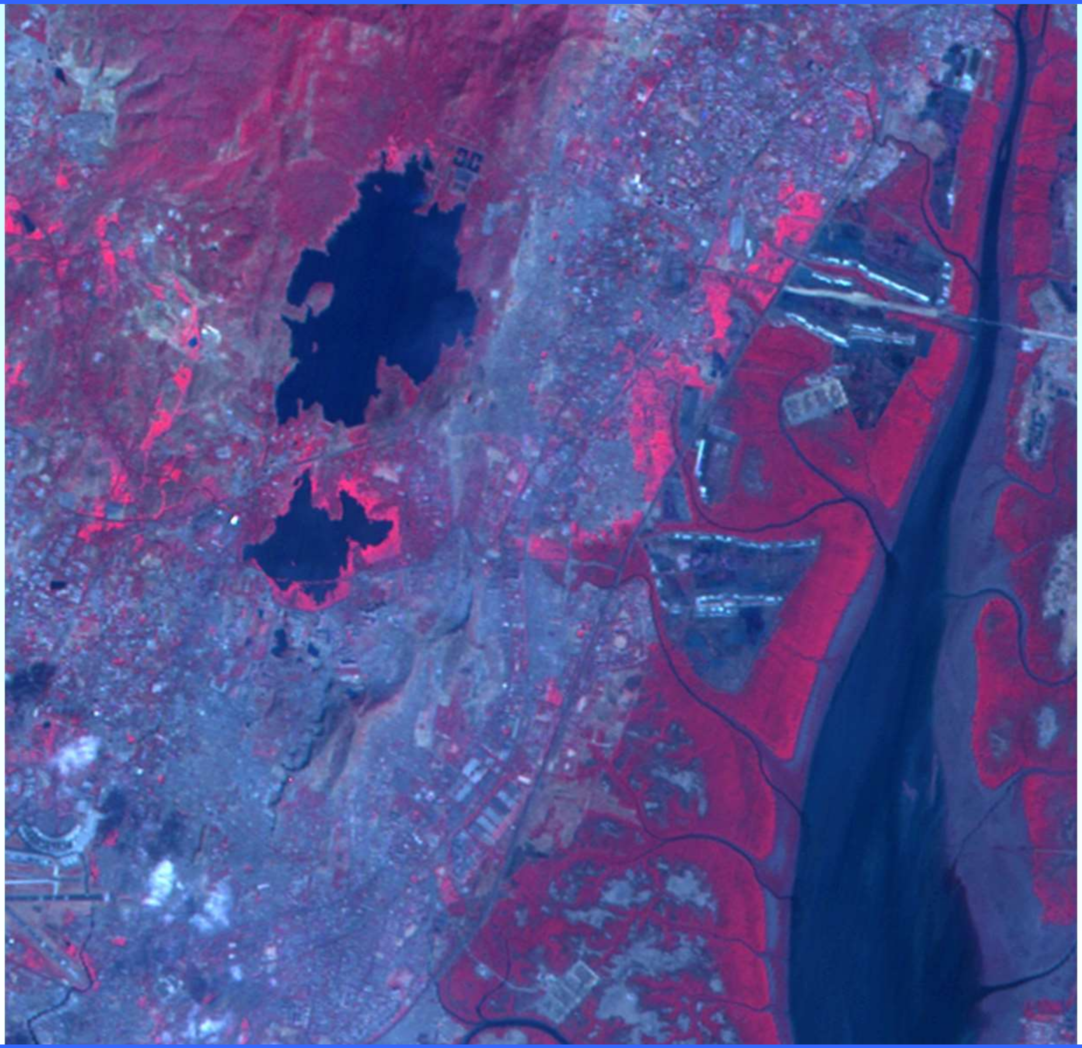
- A candidate here is the 8x8 quantization table, i.e., 64 integers ordered in the form of a linear sequence, with first eight elements repersenting 1st row of the table, second eight elements representing the 2nd row …
- In binary form, the candidate will be represented by 64x16 bits, assuming that 16 bits or $2^{16} – 1$ is the highest DN value in the image, and therefore the entries in the quantization table will also be in the same range

Input Image

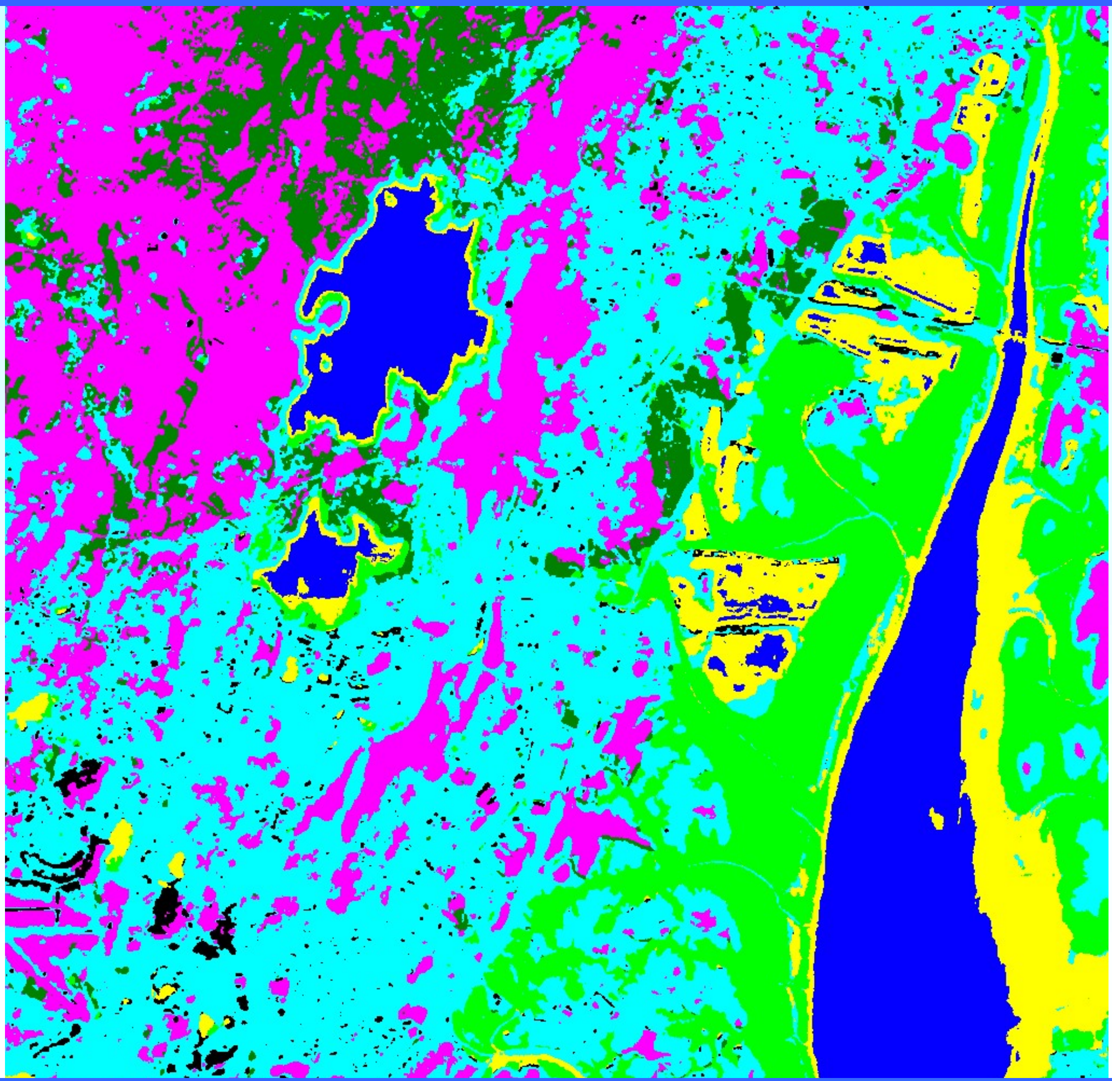
GA-NN Supervised



Example 2



Results




Water
Vegetation
Forest Vegetation
Settlements
Swamp / Marshy
Hilly Areas
Other Classes

Example: Comparison of MNF and GA results

- Class separabilities of MNF components

AVE MIN Class Pairs:
1: 2 1: 3 1: 4 1: 5 1: 6
2: 3 2: 4
2: 5 2: 6 3: 4 3: 5 3: 6 4: 5
4: 6
5: 6

Best Average Separability

1400 171 1271 1134 6531 316 838 423 1281
220 203 2811 407 171 2240 2752
395

Example: Comparison of MNF and GA results

- Class separabilities of GA components

AVE MIN Class Pairs:

AVE	MIN	1: 2	1: 3	1: 4	1: 5	1: 6	2: 3	2: 4	2: 5	2: 6	3: 4	3: 5	3: 6	4: 5	4: 6	5: 6
		Best Average Separability														
2639	337	2166	992	4767	517	1088	812	7314	3170	602	6178	509	337	4188	5270	1676

Original

CR = 6

Sample Quantization Table CR=6

2	8	9	13	13	21	22	41
8	10	12	13	21	22	40	43
10	12	15	20	23	40	44	55
11	15	20	23	39	46	54	55
17	17	25	35	46	54	56	83
17	25	35	49	51	57	63	88
28	32	50	51	59	61	104	115
30	50	50	59	61	105	112	127

Justification for GA to work

- Using a technique where we choose parents relative to their fitness (e.g. roulette wheel selection), fitter schema should find their way from one generation to another
- Intuitively, if a schema is fitter than average then it should not only survive to the next generation but should also increase its presence in the population
- If Φ is the number of instances of any particular schema S within the population at time *t*, then at *t*+1 we would expect

$$\Phi(S, t+1) > \Phi(S)$$

to hold for above average fitness schemata

Contd…