Genetic Algorithms

Based on Natural Selection

Search by
Randomized Information Exchange

Efficiently exploit
Historical Information

Originally Developed by
John Holland
University of Michigan

Classic Book
David E. Golberg
Genetic Algorithms in
Search, Optimization and
Machine Learning

Fundemental Differences

Work with a coding of the parameter set,
Not with the parameters themselves

Search a population of points,
not just one point

Use *payoff* (objective function) information
not derivatives or other aux. knowledge

Probabilistic transition Rules
not deterministic rules

Coding

The natural parameter set must be coded

A finite string of some alphabet

Each element of this string
corresponds to some information unit
with possible values

The meta optimization process
is to optimize the coding into such a string
(non-trivial)

Population of Points

One does not move from one point to the next

One starts with a population
Several points in the (coded) space
as initial guess

Moves from one population
to another

One way of avoiding local minima

Probabilistic

A randomized search

Similar to Simulated Annealing

Bad points are not thrown away

Global Optimization

Another structure to guessing the next point

Non-local (due to population)

An Example

Optimization of the function:
$f(x) = x^2$
(from 0 to 32)

The range 0 to 32 is coded in binary
5 bits

The **payoff** function
$f(x) = x^2$

Generation of Initial Points

Randomly generate
the initial population For each bit
Flip a coin

After 20 coin tosses
Generate 4 initial guesses

Calculate the payoff for each

x	coding	fitness
13	01101	169
24	11000	576
8	01000	64
19	10011	361

Operators

Reproduction
Replicate one of the elements in the population

Crossover
Exchange sections between element pairs

Mutation
Change a string element

Reproduction

Elements of the population are
Replicated

The chance of replication is based on
fitness

The more fit the element
the better chances for replication

x	coding	fitness	Prob.
13	01101	169	.144
24	11000	576	.492
8	01000	64	.055
19	10011	361	.309
total		1170	1.0

Emphasis is put on fit elements
(but not exclusively)

Samples are chosen for the other operators
on the basis of their fitness

Cross Over

A randomizing step

Parts are exchanged between elements

Two strings (population elements) are chosen

They are lined up bit by bit

A point is chosen in the string
dividing the strings in two parts

The **first** part of the first string
and
the **last** part of the last string
are merged to form a new element

The **last** part of the first string
and
the **first** part of the last string
are merged to form a new element

Crossover Example

Two elements are chosen from the population

01101
11000

A point is chosen to crossover

01 | 101
11 | 000

New elements are formed

11101
01000

These are added to the population

Crossover: Random Selection

Which two elements?

based on the fitness as above

x	coding	fitness	Prob.
13	01101	169	.144
24	11000	576	.492
8	01000	64	.055
19	10011	361	.309

Where to crossover?

a position is selected

A number between 1 and 4

1	between bits 0 and 1
2	between bits 1 and 2
3	between bits 2 and 3
4	between bits 3 and 4

Crossover: Exchange of Information

Randomness of exchange (allowing uphill climbing in search space)

Each substring
a notion or idea
within the entire concept
(the entire string)

crossover is the exchange of ideas

Good ideas can be combined with other good ideas from others

The fitness function promotes the rewarding of good combinations

Mutation

A second randomizing mechanism

Not performed very often

Another Uphill climb

This would be similar (in principle) to the monte carlo or simulated annealing principles

The decision to accept or reject is based on its status in the population

Mutation

A element is chosen
according to its fitness

A position is chosen

The position is **mutated**
to one of its possible values

One Cycle

x	coding	fitness	Prob.
13	01101	169	.144
24	11000	576	.492
8	01000	64	.055
19	10011	361	.309
total		1170	1.0

After Reproduction

x	coding	fitness	Prob.
13	01101	169	.144
24	11000	576	.492
24	11000	576	.492
19	10011	361	.309
total		1170	1.0

Due to the low probability
01000 was ignored

Due to high probability
11000 was duplicated twice

Cross Over

x	coding	mate	Prob.
13	01101	2	4
24	11000	1	4
24	11000	4	2
19	10011	3	2
total		1170	1.0

From

11 | 000

10 | 011

New elements are formed

11011

10000

From

0110 | 1

1100 | 0

New elements are formed

01100

11001

Cross Over

x	coding	fitness.
12	01100	144
25	11001	625
27	11011	729
16	10000	256
total		1754

Fitness Improvement
$1170 \rightarrow 1754$

Maximum improvement
$576 \rightarrow 729$

Terminology

chromosome	string
gene	feature,character
allele	feature value
locus	string position
genotype	structure
phenotype	parameter set
epistasis	non-linearity