Genetic Algorithm
遗传算法

Probably everybody’ s GA is unique! Many variations in population size, in initialization methods, in fitness definition, in selection and replacement strategies, in crossover and mutation are obviously possible.

By 吴婷

Contents

Common Ideas

genetic algorithm

1975 By John Holland *Adaptation in Natural and Aritificial Systems*

evolutionary computing or evolutionary algorithms

evolutionsstrategie

1960s By Ingo Rechenberg and Hans-Paul Schwefel

focussed almost exclusively on 'mutation' operators which are generally closer in concept to the types of operator used in neighbourhood search and its extensions.

evolutionary programming

1960s by Bremermann, Fogel and others

- The common thread in these ideas was the use of mutation and selection —the concepts at the core of the neo-Darwinian theory of evolution.
- GA had an extra ingredient—the idea of recombination.
- GA can be used to solve a discrete or combinatorial optimization problems.

Why GA works

Holland's explanation

- the re–combination of small pieces of the genotype (good schemata)into bigger pieces.
- intrinsic (or implicit) parallelism— information on many schemata can be processed in parallel.
- * Schema Theorem – if there are N(S, t) instances of a given schema S in the population at time t, then at the next time step (following reproduction), the expected number of instances in the new population can be bounded.

“ # Basic Conceptions

objective function

$f: \chi \to R$ χ is a discrete searching space, f is a function.

‘phenotype’

descrete search space which consists of vectors of decision variables.

‘genotype’

It is the encoded representation of the variables, usually presented as a string of length l made up of symbols drawn from an alphabet A , using a mapping $c: A^l \to \chi$

It is also presented as chromosome.

Encoding scheme(Gene Representation) is aimed to encode the chromosome, which is the most important and basic step in the algorithm.

Basic Conceptions

algorithm template

Genetic operators

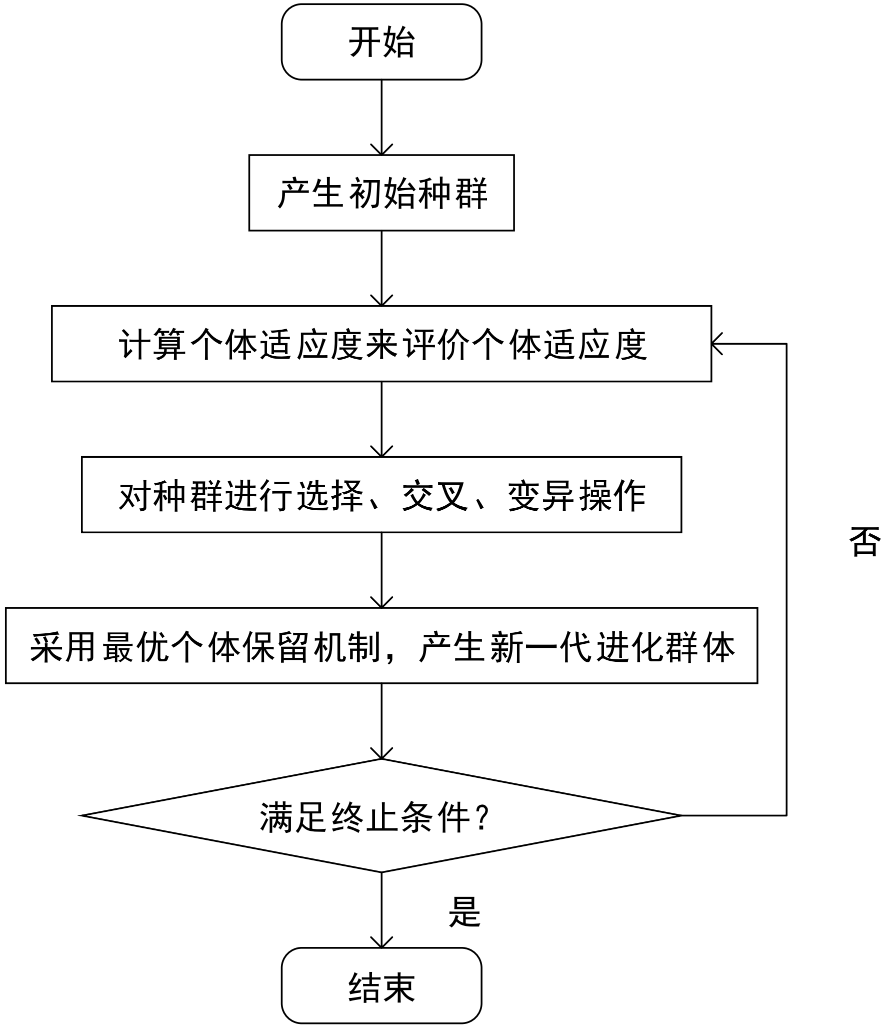
Evaluate

```
Choose an initial population of chromosomes;
while termination condition not satisfied do
  repeat
    if crossover condition satisfied then
    {select parent chromosomes;
    choose crossover parameters;
    perform crossover};
    if mutation condition satisfied then
    {choose mutation points;
    perform mutation};
    evaluate fitness of offspring
  until sufficient offspring created;
select new population;
endwhile
```

Get the best solutions

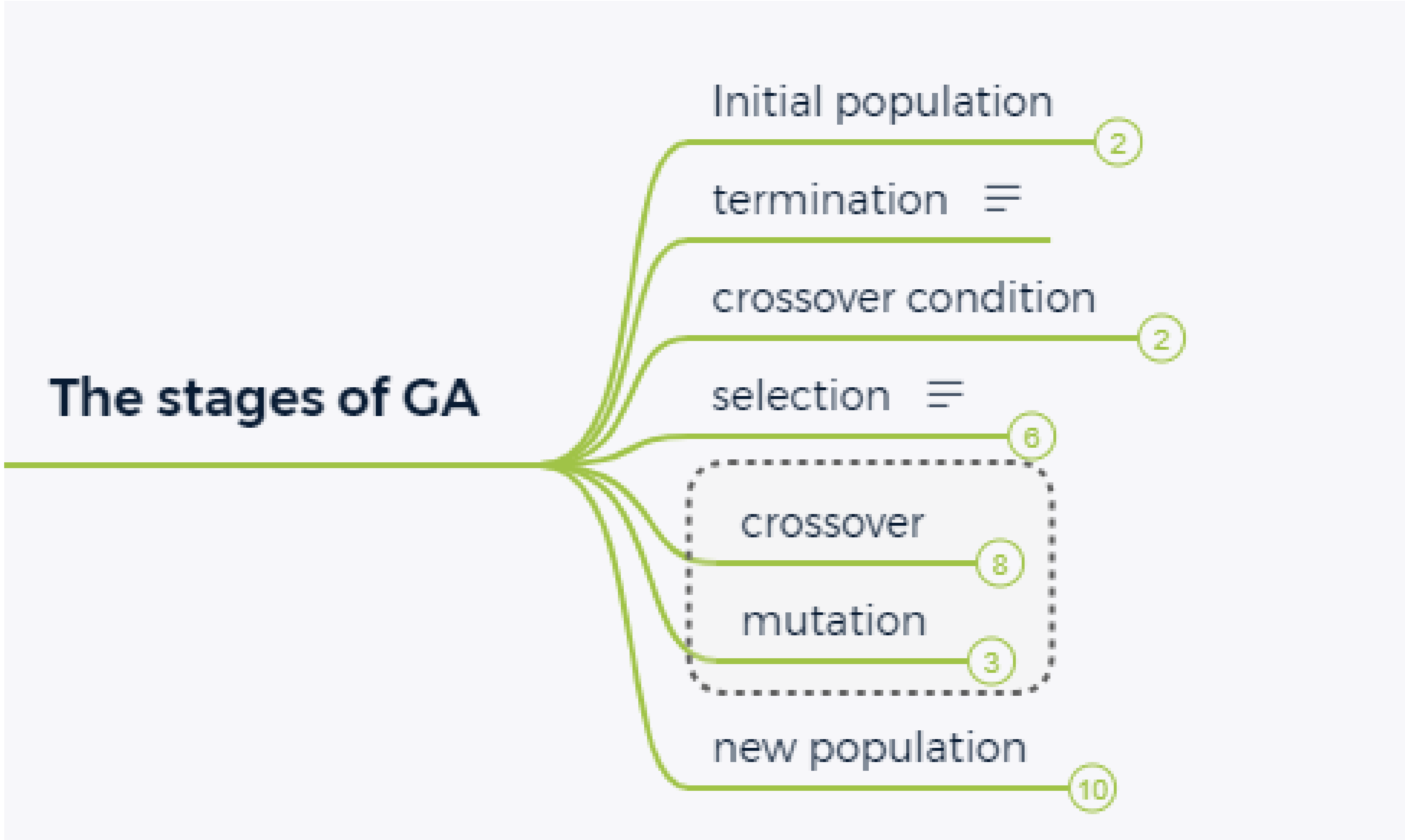
Construct a fitness function for the objective function of the problem. Fitness decides whether the gene can be preserved to the next generation.

遗传算法基本流程



开始
产生初始种群
计算个体适应度来评价个体适应度
对种群进行选择、交叉、变异操作
采用最优个体保留机制，产生新一代进化群体
满足终止条件？
否
是
结束

The Stages of GA



Initial population

- **the size of the population**

  population size(NP), generally be set to [100,1000]
- **the method by which the individuals are chosen**

Termination

GAs are stochastic search methods that could in principle run for ever. In practice, a termination criterion is needed;

Common approaches are to **set a limit on the umber of fitness evaluations or the computer clock time, or to track the population’ s diversity and stop when this falls below a preset threshold.** The meaning of diversity in the latter case is not always obvious, and it could relate either to the genotype or the phenotype, or even, conceivably, to the fitnesses, but the most common way to measure it is by genotype statistics.

Crossover rate

Crossover: generate offsprings by two parents.
Whether to exchange the chromosome of the parents depends on the crossover rate p_c , It is defined as the proporation of the offsprings generated by crossover to all offsprings. High p_c can attain a larger solution space ,but it also search unnecessary searching spaces which lead to expensive compution.
Crossover rate is usually set to 90%.

Mutation rate

Mutation helps to provide the gene which never appears in the population or find the lost gene during the selection.
Mutation rate p_m : the proporation of the offsprings generated by mutation to all offsprings.
Low p_m: good characteristic can' t be selected.
High p_m: leads to many random changes which makes offsprings lose the good characteristic inherited by parents.
Mutation rate is usually a small number below 5%.

crossover condition

- **crossover AND mutation**

  carry out crossover, attempts mutation on the offsprings with crossover rate x and mutation rate u.
- **crossover OR mutation**

  always do sth , either mutation or crossover, there is the further possibility of modifying the relative proportions of crossover and mutation as the search progresses.

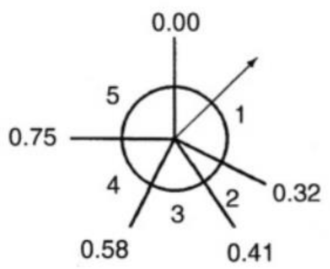
selection

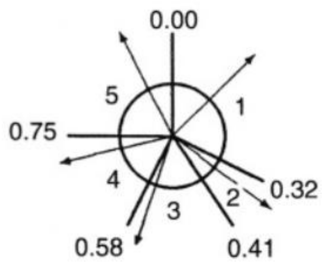
finding a suitable measure of fitness for the members of the population - scaling

Strategies roulette-wheel method(RWS)
stochastic universal selection (SUS)
Tournament Selection

Ranking the probability of selecting the string that is ranked kth in the population is denoted by P[k]. sum(P)=1
selection pressure = prob.fittest/prob.average

Selection strategies





roulette-wheel method(RWS)

uses a probability distribution for selection in which the selection probability of a given string is proportional to its fitness.

stochastic universal selection (SUS)

Instead of a single choice at each stage, we imagine that the roulette wheel has an equally spaced multi-armed spinner. Spinning the wheel produces simultaneously the values Nc for all the chromosomes in the population.

Tournament Selection

One potential advantage of tournament selection over all other forms is that it only needs a preference ordering between pairs or groups of strings, and it can thus cope with situations where there is no formal objective function at all—in other words, it can deal with a purely subjective objective function!

“

Crossover

binary replacing some of the genes in one parent by the corresponding genes of the other.

1X (one-point crossover)has no distributional bias, it limits the exchange of information between the parents.

2X(two–point crossover)

m–point crossover

UX(uniform) generating the pattern of 0s and 1s stochastically (using a Bernoulli distribution) we thus get uniform crossover.

a 1 means that the alleles are taken from the first parent, while a 0 means they come from the second

$$(a_1, a_2 | a_3, a_4, a_5, a_6)$$
$$(b_1, b_2 | b_3, b_4, b_5, b_6)$$

1X

$$(a_1, a_2 | a_3, a_4, a_5 | a_6)$$
$$(b_1, b_2 | b_3, b_4, b_5 | b_6)$$

2X

1	1	0	0	1	1
a_1	a_2	b_3	b_4	a_5	a_6

non–linear

- define an interchange mapping(in TSP)
- apply a binary mask the components corresponding to 1s are copied from one parent, and then those that correspond to 0s are taken in the order they appear from the second parent in order to fill the gaps.

Mutation

generate a binary mask by Bernoulli distribution at each locus
draw a random number for every gene in the string and compare it to u
draw a random variate from Possion distribution with parameter λ

λ : average number of mutations per chromosome. a common value for λ is 1 which means if l is the length ,mutation rate is 1/l

new population

é litism and population overlaps
ES stategies (λ , μ) preserve λ offsprings
$\lambda + \mu$ combine λ offsprings and μ parents
select members for deletion

- parents are replaced by their children needs large population and high mutation
- delete one of the worst P% of the population(rank-based)
- delete base on the age of strings

diversity maintainance

One of the keys to good performance (in nature as well as in GAs) is to maintain the diversity of the population as long as possible.

Steady-state or incremental GAs: ‘no-duplicates’. This means that the offspring are not allowed into the population if they are merely clones of existing individuals.

Booker suggested that before applying crossover, we should examine the selected parents to find suitable crossover points. This entails computing an ‘exclusive-OR’ (XOR) between the parents, so that only positions between the outermost 1s of the XOR string (the ‘reduced surrogate’) should be considered as crossover points.

1 1 0 1 0 0 1
1 1 0 0 0 1 0

Representations

- using GAs as optimizers in a search place, given a suitable encoding and fitness function
- Binary Problems: knapsap problem
- Discrete Problems : rotor stracking problem
- Permutation Problems: permutation flowshop sequencing problem(PFSP)
- non–binary problem: transform to binary string

for example:

$$f(x) = x^3 - 60x^2 + 900x + 100$$

over the search space $\chi = \{x : x \in \mathbb{Z}; x \in \{0, 31\}\}$, *i.e., the solution* x^* *is required to be an integer in the range* [0, 31].

We would use a string of 5 binary digits with the standard binary to integer mapping.

“

observations

- A steady-state (or incremental) approach is generally more effective and efficient than a generational method.
- Don’ t use simple roulette wheel selection. Tournament selection or SUS is better.
- Don’ t use one-point crossover. UX or 2X should be preferred.
- Make use of an adaptive mutation rate—one that is fixed throughout the search (even at 1/l) is too inflexible.
- Hybridize wherever possible; don’ t use a GA as a black box, but make use of any
- problem-specific information that you have.
- Make diversity maintenance a priority.
- Don’ t be afraid to run the GA several times