GENETIC ALGORITHMS

Natural Evolution:
survival of the fittest.
—— Charles Darwin

优胜劣汰，适者生存。

Outline of week 1-7:

1. GAs
2. particle swarm optimization (PSO)
3. constraint handling
4. evolution strategy (ES)
5. differential evolution (DE)
6. multi-objective evolutionary algorithms
7. multimodal optimization

★ Advantages of GAs:

1. conceptually simple
2. give a # of distinct solutions, alternative solutions are useful in planning, scheduling, multi-objective optimization, etc.
3. do not require the search space to be:
   - continuous
   - differentiable
   - unimodal

★ Simple GA (SGA):

- consider a black-box switching problem.
- 5 input switches
- output signal f(s)
- **objective**: set the switches to obtain maximum f(s)
- "on" -> 1, "off" -> 0

1 The initial population:

- assume choose 4 members in the population
- the initial population is randomly generated as:

01101, 11000, 01000, 10011

2 Fitness (objective) values:

#	String	Fitness	% of Total	# selected
1	01101	169	14.4	1
2	11000	576	49.2	2
3	01000	64	5.5	0
4	10011	361	30.9	1
Total		1170	100.0	4

3 Evolution & Genetic operations:

- reproduction / selection: applied to the current population.
- crossover: applied to the result of reproduction operation.
- mutation: applied to the result of crossover operation.

4 Reproduction:

copy solution strings from the current generation into a mating pool taking into consideration the fitness of individual strings

5 Implementation of reproduction:

- roulette wheel selection
- the wheel is spun n times to select n strings into the mating pool, n is the population size.

#	String	Fitness	% of Total	# selected
1	01101	169	14.4	1
2	11000	576	49.2	2
3	01000	64	5.5	0
4	10011	361	30.9	1
Total		1170	100.0	4

6 Crossover Operation:

- crossover probability P_c
- perform crossover $\frac{n}{2}$ times to generate n offspring
- one-point crossover is performed as follows:

① an integer position k is randomly selected along the string $1 \le k \le L-1$ where L is the string length.

e.g. 0 . 1 . 1 . 0 . 1

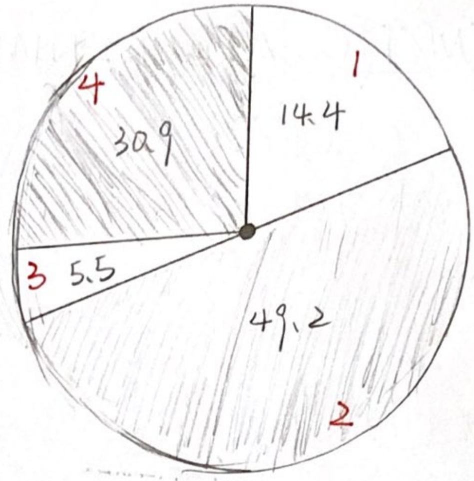
② two new strings (offsprings) are generated by breaking up the parent strings at position k.

e.g. two parent strings:

0110.1 1100.0

with k = 4, then the results of crossover operation are 2 offsprings

01100 11001



string 1: 0 ~ 0.144

string 2: 0.144 ~ 0.636

string 3: 0.636 ~ 0.691

string 4: 0.691 ~ 1.00

let the generated random numbers be: 0.127, 0.282, 0.832, and 0.496.

⇒ string 1, string 2, string 4, string 2

7 Mutation:

- randomly change the bit values in the generated offsprings

0 → 1 1 → 0

- advantage:

a) introduce a degree of diversity to the population;

b) prevent a premature convergence;

c) help to sample unexplored regions of the search space

- The mutation probability P_m is small, e.g. 0.001, 0.01.
- usually P_m is gradually reduced with the increase in the # of generations.

SGA Example: $f(x) = x^2$

Step 1: Encoding the Decision Variable:

$0 \le x \le 31$, encoded with 5 bits

Step 2: Initial Population:

01101, 11000, 01000, 10011

Step 3: Computing Fitness Proportions:

$x \Rightarrow$ maximizing $f(x) = x^2$

Step 4: Reproduction:

use the roulette wheel to obtain the mating pool

Step 5: Crossover:

1st step: select $\frac{n}{2}$ pairs of parent strings randomly

2nd step: randomly choose crossover site k

Step 6: Mutation: $P_m = 0.001$

#	initial popn.	x	f(x)	% of Total	# Selected	Mating Pool	Mate	Xover Site K	new popn.	x	f(x)
1	01101	13	169	14.4	1	01101	2	4	01100	12	144
2	11000	24	576	49.2	2	11000	1	4	11001	25	625
3	01000	8	64	5.5	0	11000	4	2	11011	27	729
4	10011	19	361	30.9	1	10011	3	2	10000	16	256
	Sum		1170	100.0	4						1754
	Average		293								439

Similarity Templates & Schemas

- a schema is a similarity template describing a subset of strings with similarity at certain string positions;
- consider {0, 1} with a wild card character $*$, create strings using alphabet {0, 1, $*$};
- e.g. schema $*101*$ represents the subset {01010, 01011, 11010, 11011};
- using schema, we can represent similarities among various strings.
- If the binary string length is 5, then there are $3^5 - 1$ different similarity templates.
- If there are k symbols in the alphabet, then there are $(k+1)^l - 1$ schemas.
- a binary string of length 5, e.g. 10111, is a member of 2^5 schemas
- a binary string of length l contains 2^l schemas.
- a population of size n and binary string of length l may have somewhere between 2^l and $n \cdot 2^l$ schemas, depending on the diversity of the population.

Similarity Templates:

The effect of reproduction:
highly fit schemas are copied more times into the mating pool.

The effect of crossover:
e.g. consider two schemas:

1 * * * 0 , * * 1 1 *

the 1st schema is more likely to be disrupted by one-point crossover than the 2nd schema.

the 1st schema is said to have a large defining length.

- The mutation rate is low and does not disrupt the schemas in a significant way.
- Observation: highly fit schemas with short defining lengths will get propagated in increasing #s.
- These highly fit schemas with short defining lengths are called building blocks.
- The # of schemas that are usefully processed in each generation is approximately n^3 where n is the population size.
- This is called implicit parallelism.

Chapter 2
Computer Implementation

Fitness functions:

SGA requires the fitness values to be positive in order to perform the roulette wheel selection or reproduction.

For maximization, if the objective function $u(x)$ can take negative values, then define the fitness $f(x)$ as:

$$f(x) = \begin{cases} u(x) + C_{min}, & \text{if } u(x) + C_{min} > 0 \\ 0 & \text{, otherwise} \end{cases}$$

C_{min} can be the absolute value of the smallest $u(x)$ (negative).

For minimization, given obj func $g(x)$, the fitness is:

$$f(x) = \begin{cases} C_{max} - g(x), & \text{if } C_{max} - g(x) > 0 \\ 0 & \text{, otherwise} \end{cases}$$

C_{max} can be the largest value of $g(x)$ in the current population.

Fitness scaling:

premature convergence:

the 1st few generations of SGA highly fit individuals may dominate the selection.

★ **Linear Fitness Scaling:**

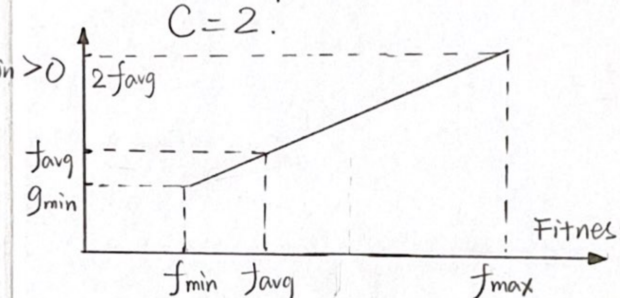
given non-negative fitness $f(x)$ the scaled function:

$$g = af + b$$

$$\Rightarrow \boxed{g_{avg} = C f_{avg}}$$

$$a = \frac{f_{avg}(C-1)}{f_{max} - f_{avg}}, \quad b = f_{avg}(1-a)$$

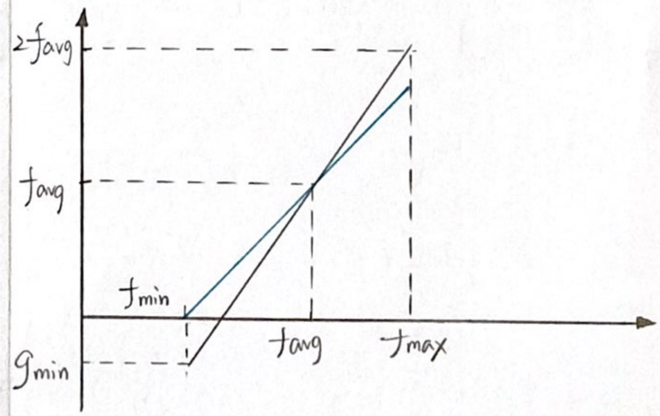
e.g. at most 2 samples in the mating pool by setting $C = 2$.



The scaling may produce negative fitness in later generations when many individuals have fitness values similar to the best individual

$\Rightarrow$ 1. maintain the average as the same before and after scaling.

2. map the min fitness to 0.



Chapter 4

Mathematical Analysis of SGA

- <u>Schema or Similarity Template</u>

- represent a subset strings with similarities at certain string positions.
- e.g. schema $*101*$ represents the following strings:
{01010, 01011, 11010, 11011}
- for {0, 1, *}, if the string is of length $l = 5$, then there are 3^5 different similarity templates, larger than the total # of strings: 2^5.
- for alphabets with k symbols, there are $(k+1)^l$ schemas.

- <u>Short highly fit schemas:</u>

- are likely to be reproduced in increasing numbers.
- are known as building block.

- Order of Schema:

If H is a schema taken from the 3-letter alphabet {0, 1, *}, then the order of H is denoted by $o(H)$, which is the # of fixed positions in the string.