

Questions 9

Dr. Roman Belavkin

BIS4435

Question 1

Give an example of combinatorial problem. What is the most difficult in solving these problems?

Answer: *One classical example is the Travelling Salesman problem (TSP), described in the lecture notes. Another example is the timetable problem. The main difficulty is that the number of combinations (and, hence, the number of possible solutions) grows much faster than the number of items involved in the problem (i.e. the number of cities in TSP, the number of time-slots, etc). This problem is known as combinatorial explosion.*

Question 2

Name and describe the main features of Genetic Algorithms (GA).

Answer: *Genetic Algorithms (GA) use principles of natural evolution. There are five important features of GA:*

Fitness Function *represents the main requirements of the desired solution of a problem (i.e. cheapest price, shortest route, most compact arrangement, etc). This function calculates and returns the fitness of an individual solution.*

Encoding *possible solutions of a problem are considered as individuals in a population. If the solutions can be divided into a series of small steps (building blocks), then these steps are represented by genes and a series of genes (a chromosome) will encode the whole solution. This way different solutions of a problem are represented in GA as chromosomes of individuals.*

Selection *operator defines the way individuals in the current population are selected for reproduction. There are many strategies for that (e.g. roulette-wheel, ranked, tournament selection, etc), but usually the individuals which are more fit are selected.*

Crossover operator defines how chromosomes of parents are mixed in order to obtain genetic codes of their offspring (e.g. one-point, two-point, uniform crossover, etc). This operator implements the inheritance property (offspring inherit genes of their parents).

Mutation operator creates random changes in genetic codes of the offspring. This operator is needed to bring some random diversity into the genetic code. In some cases GA cannot find the optimal solution without mutation operator (local maximum problem).

Question 3

Consider the problem of finding the shortest route through several cities, such that each city is visited only once and in the end return to the starting city (the Travelling Salesman problem). Suppose that in order to solve this problem we use a genetic algorithm, in which genes represent links between pairs of cities. For example, a link between London and Paris is represented by a single gene 'LP'. Let also assume that the direction in which we travel is not important, so that $LP = PL$.

- a) How many genes will be used in a chromosome of each individual if the number of cities is 10?

Answer: Each chromosome will consist of 10 genes. Each gene representing the path between a pair of cities in the tour.

- b) How many genes will be in the alphabet of the algorithm?

Answer: The alphabet will consist of 45 genes. Indeed, each of the 10 cities can be connected with 9 remaining. Thus, $10 \times 9 = 90$ is the number of ways in which 10 cities can be grouped in pairs. However, because the direction is not important (i.e. London-Paris is the same as Paris-London) the number must be divided by 2. So, we shall need $90/2 = 45$ genes in order to encode all pairs. In general the formula for n cities is:

$$\frac{n(n-1)}{2}$$

Question 4

Suppose a genetic algorithm uses chromosomes of the form $x = abcdefgh$ with a fixed length of eight genes. Each gene can be any digit between 0 and 9. Let the fitness of individual x be calculated as:

$$f(x) = (a + b) - (c + d) + (e + f) - (g + h) ,$$

and let the initial population consist of four individuals with the following chromosomes:

$$\begin{aligned}x_1 &= 65413532 \\x_2 &= 87126601 \\x_3 &= 23921285 \\x_4 &= 41852094\end{aligned}$$

- a) Evaluate the fitness of each individual, showing all your workings, and arrange them in order with the fittest first and the least fit last.

Answer:

$$\begin{aligned}f(x_1) &= (6 + 5) - (4 + 1) + (3 + 5) - (3 + 2) = 9 \\f(x_2) &= (8 + 7) - (1 + 2) + (6 + 6) - (0 + 1) = 23 \\f(x_3) &= (2 + 3) - (9 + 2) + (1 + 2) - (8 + 5) = -16 \\f(x_4) &= (4 + 1) - (8 + 5) + (2 + 0) - (9 + 4) = -19\end{aligned}$$

The order is x_2 , x_1 , x_3 and x_4 .

- b) Perform the following crossover operations:

- i) Cross the fittest two individuals using one-point crossover at the middle point.

Answer: *One-point crossover on x_2 and x_1 :*

$$\begin{array}{l}x_2 = \mathbf{8712} \mid \mathbf{6601} \\x_1 = 6541 \mid 3532\end{array} \Rightarrow \begin{array}{l}O_1 = \mathbf{8712}3532 \\O_2 = 6541\mathbf{6601}\end{array}$$

- ii) Cross the second and third fittest individuals using a two-point crossover (points b and f).

Answer: *Two-point crossover on x_1 and x_3*

$$\begin{array}{l}x_1 = \mathbf{65} \mid \mathbf{4135} \mid \mathbf{32} \\x_3 = 23 \mid 9212 \mid 85\end{array} \Rightarrow \begin{array}{l}O_3 = \mathbf{65}9212\mathbf{32} \\O_4 = 23\mathbf{4135}85\end{array}$$

- iii) Cross the first and third fittest individuals (ranked 1st and 3rd) using a uniform crossover.

Answer: In the simplest case uniform crossover means just a random exchange of genes between two parents. For example, we may swap genes at positions a , d and f of parents x_2 and x_3 :

$$\begin{array}{l} x_2 = \underline{8}71\underline{266}01 \\ x_3 = \underline{2}39\underline{212}85 \end{array} \Rightarrow \begin{array}{l} O_5 = 2\underline{7126}201 \\ O_6 = \underline{839216}85 \end{array}$$

- c) Suppose the new population consists of the six offspring individuals received by the crossover operations in the above question. Evaluate the fitness of the new population, showing all your workings. Has the overall fitness improved?

Answer: The new population is:

$$\begin{array}{l} O_1 = 87123532 \\ O_2 = 65416601 \\ O_3 = 65921232 \\ O_4 = 23413585 \\ O_5 = 27126201 \\ O_6 = 83921685 \end{array}$$

Now apply the fitness function $f(x) = (a+b) - (c+d) + (e+f) - (g+h)$:

$$\begin{array}{l} f(O_1) = (8+7) - (1+2) + (3+5) - (3+2) = 15 \\ f(O_2) = (6+5) - (4+1) + (6+6) - (0+1) = 17 \\ f(O_3) = (6+5) - (9+2) + (1+2) - (3+2) = -6 \\ f(O_4) = (2+3) - (4+1) + (3+5) - (8+5) = -8 \\ f(O_5) = (2+7) - (1+2) + (6+2) - (0+1) = 13 \\ f(O_6) = (8+3) - (9+2) + (1+6) - (8+5) = -6 \end{array}$$

The overall fitness has improved.

- d) By looking at the fitness function and considering that genes can only be digits between 0 and 9 find the chromosome representing the optimal solution (i.e. with the maximum fitness). Find the value of the maximum fitness.

Answer: The optimal solution should have a chromosome that gives the maximum of the fitness function

$$\max f(x) = \max [(a+b) - (c+d) + (e+f) - (g+h)] .$$

Because genes can only be digits from 0 to 9, the optimal solution should be:

$$x_{\text{optimal}} = 99009900 ,$$

and the maximum fitness is

$$f(x_{\text{optimal}}) = (9 + 9) - (0 + 0) + (9 + 9) - (0 + 0) = 36$$

- e) By looking at the initial population of the algorithm can you say whether it will be able to reach the optimal solution without the mutation operator?

Answer: No, the algorithm will never reach the optimal solution without mutation. The optimal solution is $x_{\text{optimal}} = 99009900$. If mutation does not occur, then the only way to change genes is by applying the crossover operator. Regardless of the way crossover is performed, its only outcome is an exchange of genes of parents at certain positions in the chromosome. This means that the first gene in the chromosomes of children can only be either 6, 8, 2 or 4 (i.e. first genes of x_1 , x_2 , x_3 and x_4), and because none of the individuals in the initial population begins with gene 9, the crossover operator alone will never be able to produce an offspring with gene 9 in the beginning. One can easily check that a similar problem is present at several other positions. Thus, without mutation, this GA will not be able to reach the optimal solution.

Question 5

What two requirements should a problem satisfy in order to be suitable for solving it by a GA?

Answer: GA can only be applied to problems that satisfy the following requirements:

- The fitness function can be well-defined.
- Solutions should be decomposable into steps (building blocks) which could be then encoded as chromosomes.

Question 6

A budget airline company operates 3 plains and employs 5 cabin crews. Only one crew can operate on any plain on a single day, and each crew cannot work for more than two days in a row. The company uses all planes every day. A Genetic Algorithm is used to work out the best combination of crews on any particular day.

- a) Suggest what chromosome could represent an individual in this algorithm?

Answer: On each day, a solution is a combination of 5 cabin crews assigned to 3 airplanes. Thus, we could encode different crews by different genes:

Genes (crews) : A, B, C, D, E

and then a chromosome representing a solution would be a chain of 3 genes (i.e. 3 cabin crews):

Solution 1 : $A B C$
 Solution 2 : $A B D$

 Solution 10 : $C D E$

In this case, each position in the chromosome corresponds to a plane. Another encoding is possible for the same problem. For example, we could encode if a crew is assigned on a plane by number 1 (if it does not matter which plane it is) and 0 if the crew has a day off:

Genes (work, day off) : 1, 0

Then we can use a chromosome of 5 genes each corresponding to a particular crew with 0 or 1 representing if a crew is at work or has a day off:

Solution 1 : 1 1 1 0 0
 Solution 2 : 1 1 0 1 0

 Solution 10 : 0 0 1 1 1

As you can see, the same problem can be encoded in many different ways.

- b) Suggest what could be the alphabet of this algorithm? What is its size?

Answer: This depends on encoding used. In the first case, when genes represent the crews, the alphabet consists of 5 letters. In the second case, when binary representation is used, only two genes are required.

- c) Suggest a fitness function for this problem.

Answer: You may come up with different versions, but it is important for the fitness to take into account the condition that cabin crews cannot work more than 2 days in a row. For example, the fitness function can take into account how many days each crew has left before a day off (e.g. 1 or 0). The fitness could be calculated as the sum of these numbers for all drivers in the chromosome. So, if d is the number of days an employee has before a day off, and m is the number of employees in the selected crews, then the fitness can be computed as

$$\text{Fitness} = d_1 + d_2 + \cdots + d_m = \sum_{i=1}^m d_i$$

- d) How many solutions are in this problem? Is it necessary to use Genetic Algorithms for solving it? What if the company operated more planes and employed more crews?

Answer: The number of solutions is the number of times 3 crews can be selected out of 5 without replacement and without taking into account their order. The first crew can be selected in 5 different ways, the second in 4 ways and the third in 3 different ways. These numbers multiplied together will give us total number times how 3 crews can be selected randomly out of 5: $5 \times 4 \times 3 = 60$ times. However, there are 6 possible combinations in which 3 crews can be ordered, and because the order does not matter the answer is $60/6 = 10$. Thus, there are 10 possible solutions for this problem.

It is not really necessary to use GA for a problem with such a small population, because solutions can be checked explicitly. However, as the number of crews and airplanes increases, so does the number of solutions, and the use of GA can be the only option. In fact, if n is the number of cabin crews and $k \leq n$ is the number of airplanes, then the number of solutions is

$$\frac{n!}{k!(n-k)!}$$

For example, if the company operated 10 airplanes and employed 20 cabin crews, then the number of solutions would be

$$\frac{20!}{10!(20-10)!} = 184,756$$