ISSN 1733-8670

ZESZYTY NAUKOWE NR 9(81)
AKADEMII MORSKIEJ
W SZCZECINIE

INLAND SHIPPING 2005

Stanisław Iwan

An Idea of Using Genetic Algorithm for Solving the Problem of River Ports Location

Key words: genetic algorithms, optimization, localization problem, transportation problem

Genetic algorithms are a very interesting optimization method, which use the natural selection idea for the optimal decision taking. These methods are usable for the solution taking in non-deterministic problems and with incomplete knowledge of the decision situation. River ports localization seems to be an interesting optimization problem of inland shipping organizing. It is necessary to find the places for ports with the lowest cost of goods distribution center localization problem and balanced or non-balanced transportation problem. This paper is focused on the idea of using genetic algorithm for solving it.

Koncepcja wykorzystania algorytmu genetycznego w rozwiązywaniu problemu lokalizacji portów rzecznych

Słowa kluczowe: algorytmy genetyczne, optymalizacja, problem lokalizacji, zagadnienie transportowe

Algorytmy genetyczne są bardzo interesującą metodą poszukiwania rozwiązań, w której w celu wyboru decyzji optymalnej wykorzystywana jest koncepcja doboru naturalnego. Metoda ta jest przydatna do rozwiązywania problemów niedeterministycznych oraz w sytuacjach decyzyjnych, w których dysponuje się wiedzą niepełną. Problem lokalizacji portów rzecznych wydaje się być ciekawym problemem optymalizacyjnym z zakresu organizacji śródlądowego transportu wodnego. W problemie tym konieczne jest znalezienie takiego położenia dla portów, aby całkowity koszt dystrybucji dóbr do poszczególnych odbiorców był jak najmniejszy. Zagadnienie to można potraktować jako połączenie problemu lokalizacji centrum dystrybucji oraz zagadnienia transportowego. Opracowanie niniejsze skoncentrowane jest na przedstawieniu idei zastosowania algorytmu genetycznego do rozwiązywania tegoż problemu.

Simple genetic algorithms

Genetic algorithms are inspired by nature. Growing the best-adapted individuals is made by the natural selection and crossover method using the genetic structure of those individuals (structure of chromosomes). This idea is the root of genetic algorithms.

In this optimization method every solution is represented by the genes sequence (coding sequence). It is an equivalent of a chromosome (so it is usually called "chromosome"). This structure is constructed by a symbol string. Every value in this string represents a value from definite value range (values of gene are called "allele"). An example of the simple chromosome is illustrated below (fig. 1).

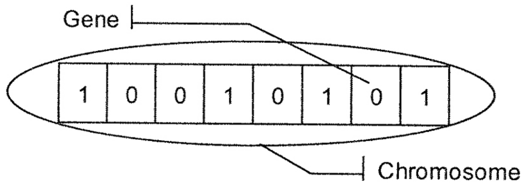

Figure 1. Simple chromosome with binary alleles

The collection of chromosomes represents the genotype of an individual (a lot of applications used genotypes built of only one chromosome). The genotype describes an individual in the coding space, but the environment needs something else for evaluation of the individual properties. In the solutions individual space is represented by phenotype, which is coded by genotype (in other words: the genotype describes the structure of genes, the phenotype describes effects of that structure). Because of that reason a small change in genotype structure can bring big changes in the phenotype, example – fig. 2.

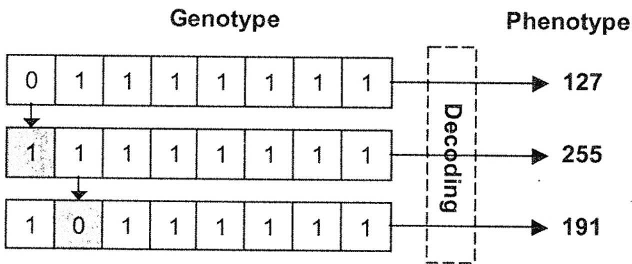

Figure 2. Relation: little change of genotype – big change of phenotype

The idea of the optimal decision taking in genetic algorithms is based on the selection of the best individuals from the actual population, crossover genes among them and mutation of the random selected genes in random selected individuals (fig. 3).

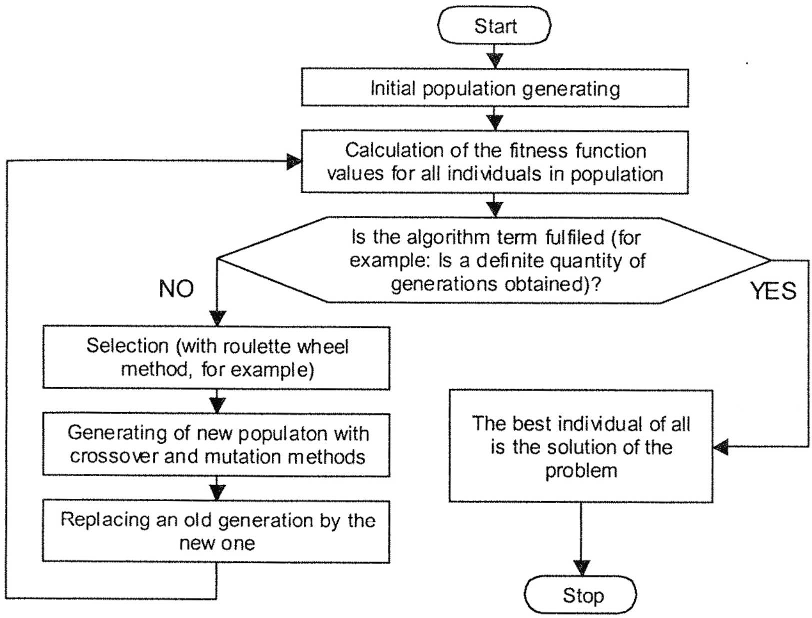

Figure 3. Simple genetic algorithm

The key for the selection is the fitness function of an individual. The fitness function permits to select the individuals with the best genotype of all population. The structure of that function is relative to the specific optimization problem. The principal selection method is the roulette wheel selection: chance that x_i individual will be selected is calculated as [5]

$$\frac{f(x_i)}{\sum_{j=1}^{n} f(x_j)},$$

where n is the number of individuals in the population.

When the individuals are selected, crossover point is random calculated. After that, parts of genes from this point in two parents' chromosomes are changed (fig. 4).

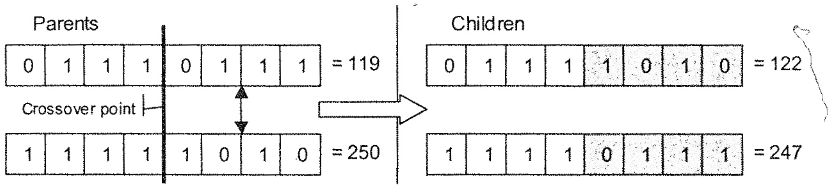

Figure 4. Binary crossover

Mutation is the special method, which can add some changes to the current population. By random choosing there is one gene (or more) selected and then his allele is changed to the reverse one (fig. 5).

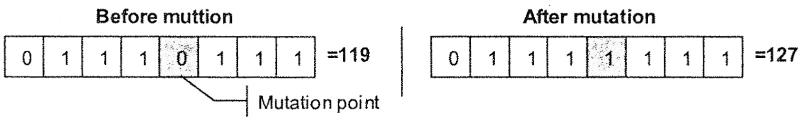

Figure 5. Binary mutation

The genetic algorithm presented above is a simple, elementary example, which uses the elementary signs set (a binary alphabet) including only two signs: 0 and 1. Many times there is a need to use a more complicated structure of the chromosome (sometimes the genotype should be built from more chromosomes), kind of crossover and mutation methods. Therefore a very important aspect in the more difficult problems is the values representation in the chromosomes. There are three principal types of the value representation in genetic algorithms [5]:

- binary – allele comprise only two values: 0 or 1,
- integer – allele comprise integer values from defined range,
- real-valued – allele comprise real values from defined range with defined precision.

The binary representation is the most popular method of encoding, used in many applications. There is a group of problems which can use only binary values in the fitness function (binary linear programming, for example: machine using planning). Then the chromosome length is adequate to the quantity of variables. When variables in the fitness function (determined by optimization problem structure) are integer or real-valued, there are two possibilities:

- use integer or real-valued representation,
- convert these values to binary representation.

River ports location problem

The river port location problem is similar to the supply or distribution point location problem. It is necessary to find a place for the distribution center, where it is possible to distribute goods from this point to the purchasers at the lowest cost. If the transportation rate and goods quantity are known, it is necessary to use a co-ordinate system and indicate the co-ordinates of all purchasers points in it. The total cost of distribution from the distribution center to every purchaser can be calculated as:

$$C = \sum_{i=1}^{I} r_i \cdot q_i \cdot d_i,$$

where I – the purchasers quantity; r_i – transport rate to i purchaser; q_i – goods quantity for i purchaser; d_i – distance between center and i purchaser (fig. 6):

$$d_i = \sqrt{(x_i - x_C)^2 + (y_i - y_C)^2} \; .$$

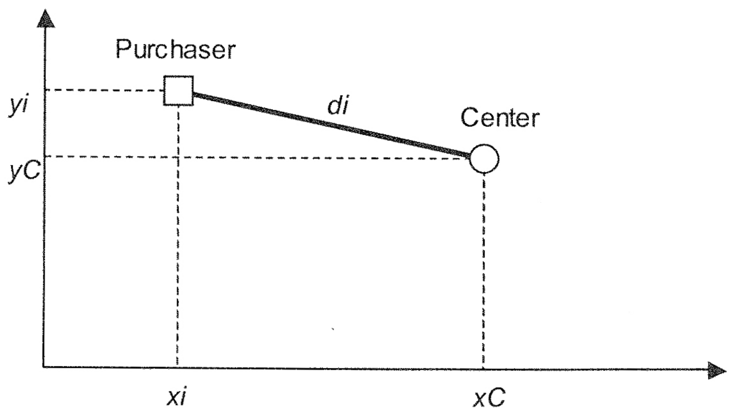

Figure 6. Distance between center and purchaser in co-ordinate system

In a simple situation, with only one center to localize, there is one important question: where should the distribution center be localized; it means: what co-ordinates in the co-ordinates system it has to have when compared to the purchaser's positions. But if there are more centers to localize it is necessary to add another one: how many goods should be distributed from every point and where (to which purchaser or purchasers)?

The river port is a kind of distribution center with the special property: one of co-ordinates is known (it is a river location). So there is necessity to find another one and a goods distribution net (if there are more than one port). The optimization problem is to find the distribution network with the lowest cost, and it can be calculated in the following way:

$$\sum_{i=1}^{I}\sum_{j=1}^{J} r_{i,j} \cdot q_{i,j} \cdot d_{i,j} \to \min,$$

where:

- I – ports quantity;
- J – purchasers (let's call it cities) quantity;
- $r_{i,j}$ – transport rate for distribution from i port to j city;
- $q_{i,j}$ – goods quantity distributed from i port to j city;
- $d_{i,j}$ – distance between i port and j city.

There are two kinds of constrains, similar to the constrains in balanced transportation problem:

1. for every port (i constrains):

$$\sum_{j=1}^{J} q_{i,j} = S_i,$$

where: S_i – supply of i port.

2. for every city (j constrains):

$$\sum_{i=1}^{I} q_{i,j} = D_j,$$

where: D_j – demand of j city;

Therefore, the problem presented above can be treated as a compilation of the distribution center location problem and the transportation problem. It is easy to solve each of them separately but together it is much more difficult. The genetic algorithm seems to be very helpful for that.

Figure 7 shows this problem for two ports and four purchasers (cities).

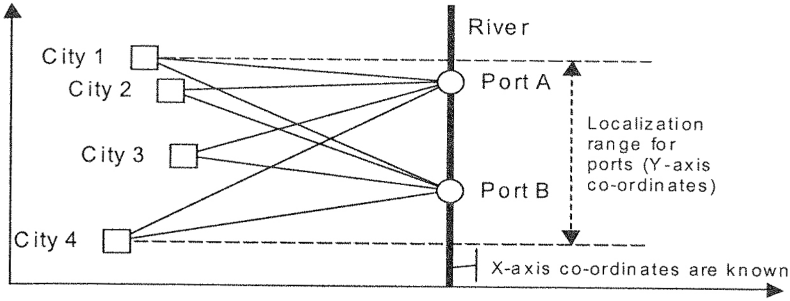

Figure 7. Distribution network: 2 ports and 4 cities

Genetic algorithm for the problem presented above

Chromosome structure

The most important thing in the genetic algorithm is the correct construction of the chromosome. The structure of chromosome is determined by assumed type of variables (binary, integer or real) and value range for them. There are two kinds of variables in the problem presented above:

- Y-axis co-ordinate of port – it is a real type, one for every port;
- goods quantity distributed from port to city – it is a integer type, one for every connection between every port and every city.

Every variable will be represented in the chromosome by one gene with real or integer values. A structured chromosome will be parted to segments: every port with every connection between this port and every city. For the example presented in figure 7, the chromosome structure will have two parts with one Y-axis co-ordinate and four goods quantities for every one of them (fig. 8).

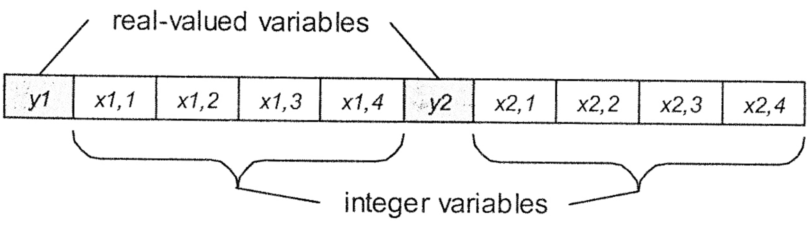

Figure 8. Structure of chromosome for 2 ports and 4 cities

In the next step it is important to assign correct value range for the variables. Therefore, let:

- the lowest value for Y-axis co-ordinate of every port (y_1 and y_2) be equal to the lowest Y-axis co-ordinate for cities;
- the highest value for Y-axis co-ordinate of port (y_1 and y_2) be equal to the highest Y-axis co-ordinate for cities;

and

- the goods quantity distributed to the j city ($x_{1,j}$ and $x_{2,j}$) be an integer from 0 to demand value of that city.

Fitness function

The fitness function is determined by the optimal decision criterion (objective function) of the problem. Sometimes forms of both functions can be the same but many times there is a need of conversion. The idea of genetic algorithms is based on two assumptions [2, 3]:

- fitness function can take only non-negative values – it is important for the selection,
- fitness function is the maximization function – the best individuals are chosen (with the best fitness values).

For that reason maximization problems are preferred for genetic algorithms. For minimization problems there are two methods of conversion. The easiest way is based on duality of optimization problems. The fitness function is calculated as:

$$f(x) = -1 \cdot g(x),$$

where: $f(x)$ – fitness function, $g(x)$ – objective function. But it is a good method only for problems with non-negative values. The best method of objective function conversion to the fitness function form, which includes non-negative values in objective function is [2]:

$$f(x) = \begin{cases} C_{\max} - g(x) & \text{for } g(x) < C_{\max} \\ 0 & \text{for others} \end{cases},$$

where: C_{max} – constant.

It is possible to assign the value for constant C_{max} by means of four methods [3]:

- *a priori*,
- as the highest $g(x)$ value in current population,
- as the highest $g(x)$ value in all the populations including the current one,
- as the highest $g(x)$ value in the last n populations.

The problem presented above includes only non-negative values and has a minimization objective function. It is possible to use the easy conversion method and the fitness function will be an inversion of the objective function.

Constraints

There are three basic methods of constrains including genetic algorithms [1, 2]:

- penalty functions,
- repair algorithms,
- special genetic operators and detectors use.

For the above problem, there can be used a simple repair algorithm, but first it is important to make a little change in this problem. Let the supply be unknown, then the amount of constrains will be reduced and will include only one kind of them – for every city:

$$\sum_{i=1}^{I} q_{i,j} \leq D_j .$$

The repair algorithm for the problem with that change is presented below.

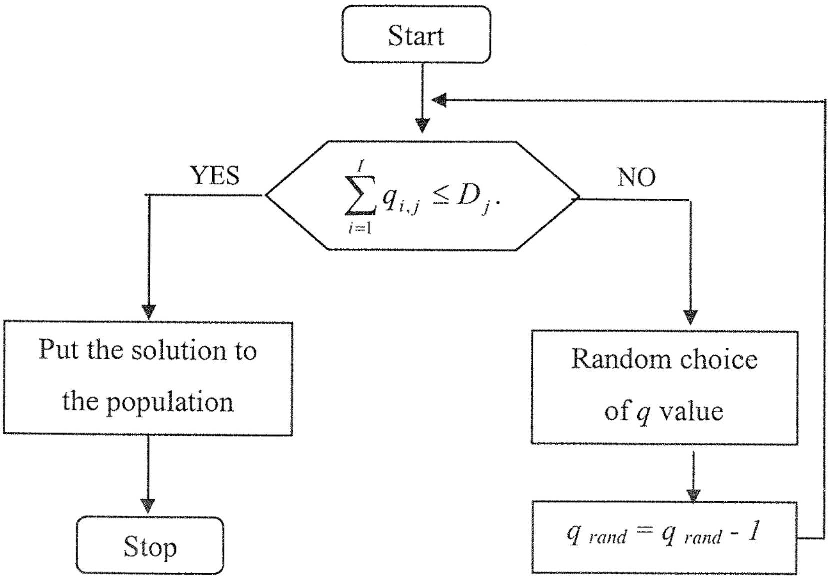

Figure 9. Repair algorithm for the above problem

In this situation it is important to include the cost of not realized demands into the objective function:

$$\sum_{i=1}^{I}\sum_{j=1}^{J} r_{i,j} \cdot q_{i,j} \cdot d_{i,j} + \left(\left(\sum_{j=1}^{J} D_j - \sum_{i=1}^{I} S_i\right) \cdot p\right) \rightarrow \min,$$

where: p – penalty cost of not delivered goods.

Genetic operators

For the above problem it is possible to use simple genetic operators [4]:
- standard proportional selection for maximization problems (based on roulette wheel selection) incorporating elitist model – makes sure that the best member survives,
- one point crossover,
- random uniform mutation (a variable selected for mutation is replaced by a random value between lower and upper bounds of this variable).

Conclusion

This paper was focused on a proposal of genetic algorithms used for optimization in inland shipping. The river ports location problem seems to be a very interesting example for the exploration. The above proposal includes only basic aspects of that. There are a lot of other problems to resolve like the curve of the river (every port has the same X-axis co-ordinate above but usually this value will be variable) or location of ports on both sides of the river. The aim of this paper was to present genetic algorithms as the optimization method, good for many problems of inland shipping arrangements.

Summary

Genetic algorithms appear to be a very interesting optimization method, which makes use of the natural selection idea for the optimal decision taking. These methods are usable for the solution taking in non-deterministic problems and in incomplete (less-parametric) decision situation. The river ports location seems to be an interesting optimization problem of the inland shipping organizing. It is necessary to find the places for ports with the lowest cost of goods distribution from ports to the purchasers. This problem can be treated as a compilation of the distribution center location problem and the balanced or non-

balanced transportation problem. This paper is focused on the idea of genetic algorithm used for solving it.

References

1. Arabas J., *Wykłady z algorytmów ewolucyjnych*, WNT, Warszawa 2004
2. Goldberg D. E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, Addison-Wesley 1995
3. Gwiazda T. D., *Algorytmy genetyczne. Zastosowanie w finansach*, Wydawnictwo Wyższej Szkoły Przedsiębiorczości i Zarządzania im. L. Koźmińskiego, Warszawa 1998
4. Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Berlin Heidelberg 1996
5. Rothlauf F., *Representations for Genetic and Evolutionary Algorithms*, Springer-Verlag, Berlin Heidelberg 2002
6. Schaefer R., *Podstawy genetycznej optymalizacji globalnej*, Wydawnictwo Uniwersytetu Jagiellońskiego, Kraków 2002

Wpłynęło do redakcji w grudniu 2005 r.

Recenzent
prof. dr hab. inż. Jan Kulczyk

Adres Autora:
Stanisław Iwan, Ph. D.
Instytut Zarządzania Transportem
Wydział Inżynieryjno-Ekonomiczny Transportu
Maritime University of Szczecin
e-mail: stan.ivan@wp.pl