Research and development of methods to select the optimal location of vehicles on the parking lot

Rostyslav Kryvyy[1], Oleh Kovaliuk[2]

[1]Computer-Aided Design Department, Lviv Polytechnic National University, UKRAINE, Lviv, S. Bandera street 12, E-mail: rostyslav.kryvyy@gmail.com

[2]Computer-Aided Design Department, Lviv Polytechnic National University, UKRAINE, Lviv, S. Bandera street 12, E-mail: oleh.kovaliuk@gmail.com

***Abstract* – people often face the problem of finding the place on a parking, especially it applies to sites with multiple floors or to parking lots in large shopping centers with significant traffic. In such places drivers are forced to wander in search of a place, creating considerable inconvenience. Besides that, cars are parked in a chaotic manner and noone thinks how to optimize their location. That is why it is necessary to create a system that can simplify this process and automatically search a proper place.**

Key words – parking, parking lot, vehicles, cars, optimization, genetic algorithm.

I. The purpose of the work

The purpose of this work is research and development of an algorithm for finding the optimal location of vehicles on the parking lot, as well as its implementation. This algorithm can be used either as a part of an automated system for assigning the parking places at the entrance to the parking lot or as a part of service for reserving parking places online.

II. Explanation of input and output parameters

There are two general groups of input parameters which are needed for the algorihm. The first one includes parameters which are common for all users and should be a part of system configuration. These are mostly the data, which characterize a parking lot in general and its parking places. These parameters are described in Table 1. The second group of input parameters for development of the genetic algorithm are the data characterizing a vehicle, the time period for which it will stay on parking, and the driver's personal preferences regarding convenience and price of a parking place, all these parameters are described in Table 2.

TABLE 1

COMMON INPUT PARAMETERS

$l_1,...,l_n$	Set of all possible length values of a parking place
$w_1,...,w_n$	Set of all possible width values of a parking place
fl	Number of floors within a given parking lot
$d_1,...d_2$	Set of all possible distances from a parking place to an exit

TABLE 2

USER SPECIFIC INPUT PARAMETERS

VT	Type of vehicle, e.g. car, bus
T_s	Start date and time of parking
T_e	End date and time of parking
C	Coef value, which describes convenience and price of the place

The result of the algorithm is a set of parameters which describe the optimal location within a given parking lot, based on the input parameters specified in Table 1 and Table 2. All parameters of a final result are described in Table 3.

TABLE 3

RESULT'S PARAMETERS

l	Length of the optimal place
w	Width of the optimal place
fl	Floor on which optimal place should be located
D	Distance from the optimal place to the exit

III. Limitation of parameters

According to the physical limitations of the parking lot, parking place and the vehicle, there are a number of limitations, which should be considered within a specific parking lot:

- Minimum and maximum length of parking places within a given parking lot.
- Minimum and maximum width of parking places within a given parking lot.
- Minimum and maximum floor within a given parking lot.
- Minimum and maximum distance from a parking place to an exit.

IV. Algorithm for finding the optimal place

1. Generate random initial population considering physical limitations of a parking lot which were mentioned in Section III. Population size is 25% of the parking's size as specified in Eq. (1).

$$S_p = 0.25 \times S_{pr} \quad (1)$$

TABLE 4

EXPLANATION OF PARAMETERS FOR EQ. (1)

S_p	Size of the initial population
S_{pr}	Size of the paking lot

2. Loop through each individual within the current population created in the previous step and calculate the value of the fitness function using Eq. (3).

$$F(V) = \begin{cases} 0, \; if \; V_p < V_i \\ \dfrac{1}{\left|V_p - V_i + D \times 2\right|}, \; if \; V_p > V \end{cases} \quad (2)$$

TABLE 5

EXPLANATION OF PARAMETERS FOR EQ. (2)

F(V)	Fitness function for length or width depending on input values
V_p	Length or width of a place
V_i	Length or width of a car
D	Distance between cars

$$F = F(l) + F(w) + T \times C \times \left(\frac{1}{fl} + \frac{1}{d} \right) \quad (3)$$

TABLE 6

EXPLANATION OF PARAMETERS FOR EQ. (3)

F	Resulting fitness function
F(l)	Fitness function for length
F(w)	Fitness function for width
T	Time between the start and the end date of parking in hours
C	Coef value, which describes convenience and price of a place
fl	Floor of a place
d	Distance from a place to an exit

3. Sort individuals in order of fitness function's value, choose 50% of all individuals with the highiest value of fitness function for applying genetic operator's phase of crossing to this group.
4. For individuals selected in the previous step apply the crossing operator in the following way:
   - 4.1. Choose two random individuals from group for crossing.
   - 4.2. Each descendant inherits properties from its two parents alternately, for example the first property is inherited from the first parent, the second – from the second parent, etc.
   - 4.3. The result of a crossing operation between two individuals are two descendants which inherit properties from parents in different order.
   - 4.4. Exclude two individuals which were used during this step from the group selected for crossing.
   - 4.5. If group with individuals selected for crossing is not empty go to step 4.1, otherwise go to step 5.
5. Among the current population apply the mutation operation in the following way:
   - 5.1. Choose a random individual from the population.
   - 5.2. Choose a random property of the selected individual.
   - 5.3. Change value of the selected property to a randomly generated one.
6. Prepare new population based on individuals from the previous steps in the following way:
   - 6.1. Add to the new population all individuals which were created during the crossing operation applied in step 4.
   - 6.2. Add an individual which was created during mutation operation in step 5.
7. If last iteration is achieved, the result is an individual with the best fitness function value - if not, go back to the step 2. Number of required iterations can be calculated using Eq. (4).

$$N = 0.5 \times S_p \quad (4)$$

V. Algorithm for finding the best place on the parking lot

1. Compare all available places in the parking lot with an optimal one, using Eq. (6).

$$S(V) = \begin{cases} 0, \text{ if } V_a < V_o \\ \dfrac{1}{V_a - V_o}, \text{ if } V_a > V_o \end{cases} \quad (5)$$

TABLE 7

EXPLANATION OF PARAMETERS FOR EQ. (5)

S(V)	Similarity function for length or width depending on input values
V_a	Length or width of the available place
V_o	Length or width of an optimal place

$$S = S(l) + S(w) + \frac{1}{|fl_a - fl_o|} + \frac{1}{|d_a - d_o|} \quad (6)$$

TABLE 8

EXPLANATION OF PARAMETERS FOR EQ. (6)

S	Resulting value of similarity
S(l)	Similarity for length
S(w)	Similarity for width
fl_a	Floor of the available place
fl_o	Floor of an optimal place
d_a	Distance from the available place
d_o	Distance from the optimal place

2. A place with a largest value of the similarity function is the result of this algorithm.

Conclusion

This paper researched and developed a genetic algorithm that allows to determine the optimal positions of vehicles on the parking lot on the basis of input parameters. Also was developed a method to choose the best one among all currently available places within a parking lot. As a result of an algorithm, the real and optimal places are obtained, so we can calculate the real proximity to the ideal result and thus based on statistical data we can analyze the optimality of the algorithm as well as the structure of the parking lot and make conclusions for their improvements.

References

[1] M. Mitchell, "An introduction to genetic algorithms", The MIT Press, 1998.

[2] D. Goldberg, "Genetic algorithms in search, optimization and machine learning", Pearson Education, 2006.