The use of clonal selection algorithm for the vehicle routing problem with time windows

Marcin Ogiołda
Faculty of Automatic Control, Electronics and Computer Science
Silesian University of Technology
Akademicka 16, 44-100 Gliwice, Poland
Email: m.ogiolda@gmail.com

***Abstract*—Genetic and heuristics algorithms are suitable for finding solutions where classical methods fails. In this paper, clonal selection algorithm was discussed as a efficient solution for the vehicle routing problem with time windows. Moreover, variability of parameters has been tested, analyzed and discussed.**

I. Introduction

Vehicle Routing Problem is a combinatorial optimization problem with practical applications in logistics. It is an extension of the classic traveling salesman problem. The basic version involves the determination of optimal route passes to a group of customers for whom the demand and location are known. Delivery is carried out by a certain number of available vehicles of the same capacity route starting and finishing in the same store.

This problem occurs in several varieties. Here is a sample of its variants:

- CVRP (Capacitated Vehicle Routing Problem) – vehicles have a defined volume,
- MDVRP (Multi Depot Vehicle Routing Problem) – there are many warehouses,
- PVRP (Periodic Vehicle Routing Problem) - deliveries are spread over a number of days,
- SDVRP (Split Delivery Vehicle Routing Problem) – customers can be served by several vehicles,
- VRPTW (Vehicle Routing Problem with Time Windows) – customers have a defined time windows when they should be served.

In this study, a variant of vehicle routing problem with time windows (VRPTW) will be examined.

A. Vehicle Routing Problem with Time Windows

Vehicle Routing Problem with Time Windows (VRPTW) is a multi-criteria combinatorial optimization problem belongs to the class of NP-hard problems. The problem involves three criteria

- the number of available vehicles,
- the maximum capacity of vehicles,
- time window in which each of the clients must be served.

In addition, each customer orders varying quantity of goods, have an established individual service time and should be served exactly once. The vehicle can reach to the customer before the start of the time window in which the client should be served. However, in this case, the driver must wait until the time interval begins. The duration is also defined and it is determined individually for each client.

The solution to problem thus defined is a set of closed tracks (road vehicles) that contain all the customers exactly once and do not alter any of the accepted limits. At the same time minimize the number of used cars what is the basic criterion for optimization, and minimize the total distance traveled by all vehicles (secondary criterion optimization).

A practical example of a routing problem with time windows can be a distribution center that distributes goods to retail chains. The distribution center has its own fleet of delivery vehicles at a specified equal capacity, and individual stores have specific hours of receiving the goods. The problem is to reduce the number of required vehicles carrying the shipment of goods to the shops and to minimize the length of the route.

B. Related works

Heuristics are called methods of finding solutions to problems in acceptable time with no guarantee of finding the optimal solution. Heuristic methods are typically used to solve difficult problems for which classical approach is not able to find optimal solution in finite time. Heuristic algorithms often are inspired by the processes from different scientific fields. Evolutionary algorithms are inspired by genetics and evolution, simulated annealing algorithm was inspired by the annealing process used in metallurgy, immune algorithms are inspired by the processes occurring in the natural immune system.

Heuristic approach found numerous practical applications such as position traffic in NoSQL database systems [1]. This is a very important topic because of the growth of Internet access and the creation of various queues. Another application is signal processing like sound [2], [3] and image [4]–[8] where the idea to use a specific heuristic technique for feature selection were presented. Moreover, in [9], heuristic showed the application of heuristic to maze creation. Furthermore, optimization is important approach in engineering [10], [11], physics [12] or medicine [13] where it is used to optimize certain parameters.

CLONALG algorithm is an algorithm of the general purpose, which means that various problems can be solved by this approach, among others optimization problems [14], problems of pattern recognition [15], as well as it can be applied to machine learning [16], [17]. As part of this work CLONALG algorithm was used to solve the vehicle routing problems with time windows (VRPTW).

VRPTW is a multi-criteria combinatorial optimization problem. It has a practical application in transport. In the problem, three constraints are highlighted: the number of available vehicles, the maximum capacity of vehicles and the time windows in which each customer should be served. VRPTW is NP-hard problem and that is why heuristic approaches are used. In this paper, clonal selection algorithm as a tool for solving VRPTW is described.

II. Assumptions of the theory of clonal selection

Clonal selection is a process of specific immune response. It is the activation of only those B cells which can produce antibodies against the pathogen. Activated B cells rapidly begins to divide. The resulting clones are go through a process of somatic hypermutation, the aim of which is to produce the most relevant antibodies. Then, mutant clones are subjected to evaluation of their the degree of adaptation to a particular antigen. Clones with a high degree adaptation are converted into memory cells, or plasma, whereas clones with low degree of adjustment undergo apotheosis Plasma cells produce antibodies which combine the antigen phagocytic cells indicate that it should be absorb. The memory cells are used in the secondary immune response.

III. Clonal selection algorithm

Clonal selection algorithms is a group of immune algorithms, whose principles of operation was inspired by the theory of clonal selection in the immune system of human. There are many algorithms modeled on the theory of clonal selection, such as CLONALG, opt-IA, MOCSA, RCSA [18], [19]. In this paper, CLONALG (CLONal selection ALGorithm) was used to solve the problem of vehicle routing with time windows was used. It was presented by de Castro and Von Zubena in [19]. This algorithm can be applied to optimization problems, pattern recognition, and machine learning.

CLONALG is characterized by few rules. The first is the number of clones resulting from each antibody - the value is directly proportional to the degree of adaptation of the antibody. Whereas the resulting clones were subjected to mutation are inversely proportional to the degree of adaptation of their parents. In addition, the least matched antibodies are replaced with newly generated. Replacing the least adapted new antibodies allows you to increase the diversity of pool. In the case of optimization function allows to escape from local in order to find global extreme.

CLONALG algorithm is available in two versions. The first version is designed for pattern recognition and machine learning. The second version is adapted to optimization problems. Version of CLONALG algorithm is designed for pattern recognition and machine learning which operates on two main sets.

The first set is a set of antigens Ag which contains patterns for recognition. The second set is marked as Ab contains antibodies representing the solution to problem. Version of CLONALG algorithm for machine learning and pattern recognition can be described in a following way

1) Initiation of pool Ab of size N divided into two subsets. The first subset Ab_m of size m contains the potential of antibody memory. The second one Ab_r of size r contains other antibodies for the diversity of the population.
2) Random selection of antigen Ag_j for the purposes of recognition.
3) Calculating the degree adaptation of each antibody to the antigen Ag_j.
4) The choice of n fittest antibodies.
5) Cloning of selected antibodies directly proportional to the degree of adaptation.
6) Maturation of set of clones by mutation inversely proportional to the degree of adaptation.
7) Determination of the affinity of each clone relative to the preselected antigen.
8) Selection of the mature clone with the highest adaptation to the antigen Ag_j as a candidate for a set of memory. If its degree of adaptation with respect to the Ag_j is better than the weakest one in the set memory, it is replaced.
9) Replacement d antibodies from a pool Ab_r with the lowest adaptation to the antigen Ag_j by new antibodies.

Steps 2-9 are performed by a predetermined number of iterations specified by N_{gen}. Each iteration is synonymous with a new generation of antibodies. The algorithm returns a set of memory Ab_m.

Version of CLONALG algorithm, designed to optimize tasks is slightly different from the version used for pattern recognition and machine learning. These differences are as follows

- In the first step, the two subsets Ab_m and Ab_r are not created. The entire population is a set of memory, and each antibody represents a part of the input space.
- In the second step, the population of antigen recognition is replaced by the objective function, which is subject to optimization. The degree of adaptation of the antibody corresponds to the value of the objective function.
- In the eight step, instead of a single antibody, n antibodies are chosen to the set Ab.

The basic parameters of CLONALG algorithm are

- N - size of the pool of antibodies Ab,
- n - the number of selected best-fit antibodies,
- d - the number of the least adapted of antibodies that will be replaced by new ones,
- β - coefficient of cloning,
- N_{gen} - number of generations of antibodies (iteration of the algorithm).

The number of clones that will be created from each of n selected antibodies for cloning is defined as

$$N_c = round\left(\frac{\beta \cdot N}{i}\right) \tag{1}$$

where i is the index of antibody inversely proportional to the degree of adaptation (antibodies are ordered with respect to descending adaptation).

The number of all created clones for mutations N_t is calculated as

$$N_t = \sum_{i=1}^{n} N_c. \tag{2}$$

The formula for the number of clones may be presented in other way like

$$N_c = \left\lfloor \frac{\beta \cdot N}{i} + 0.5 \right\rfloor . \tag{3}$$

In addition, when the optimization process applies to multimodal function it is recommended that $N = n$ and changing the mathematical formula for the number of clones that will be created from each of n chosen antibodies. The modified formula is as follows

$$N_c = round\left(\beta \cdot N\right). \tag{4}$$

In CLONALG algorithm, antibodies are subjected to mutation inversely proportional to the degree of their adaptation. One possibility for the implementation of such a mutation is to introduce mutation rate α. The best-known formula for the mutation are determined as follows

$$\alpha = e^{-\rho \cdot f}, \tag{5}$$

or

$$\alpha = \left(\frac{1}{\rho}\right) e^{-f} \tag{6}$$

where

- α – mutation rate,
- ρ — aspect of ratio distribution,
- f — the value of the relevance of a given antibody normalized to the interval $\langle 0, 1 \rangle$.

The number of mutations that will be performed on the antibody is defined as

$$N_m = \lfloor \alpha \cdot L \rfloor \tag{7}$$

where L is the length of antibody that describes a certain number of attributes. For example, the traveling salesman problem the length of antibodies will be equal to the number of cities that must be visited.

Version of CLONALG adapted to optimization problems is presented in Fig. 1. There are also modifications of algorithm, CLONALG such as parallel version (called. parallel CLONALG) or CLONCLAS (CLONal selection algorithm for CLASsification). These versions have been described in detail in [18].

IV. Research on the influence of different parameters on the obtained results

In order to compare the quality of the results obtained by clonal selection in the problem VRPTW with other algorithms, a set of test data must be selected. The kit should be chosen so that the best solutions are known and publicly available. One of these data sets has been prepared by Marius Solomon. Solomon prepared three different data sets for 25, 50 and 100 clients. For the purposes of the experiment, a set composed of 100 clients was used. This set includes 56 test problems that have been divided into six distinct groups C1, C2, R1, R2, RC1 and RC2. In addition, groups are diverse in terms of the capacity of vehicles and the width of the time windows. Particular groups differ in the arrangement of clients. In the group C1, customers make explicit distinct of clusters, the group C2, the boundaries of clusters are less clear. The groups R, the distribution of customers is random throughout the area. In contrast, the group RC groups are a combination of C and R.In addition, groups are diverse in terms of the vehicles capacity and the width of time windows.

In groups C1, R1 and RC1 vehicles have a relatively low capacity and time windows are narrow. In the groups C2, R2 and RC2 capacity vehicles are larger, and the time windows are wider.

The distance between two customers x_1 and x_2 is measured by Euclidean metric defined as

$$d(x_1, x_2) = \sqrt{\sum_{i=0}^{2} (x_{1,i} - x_{2,i})^2}. \tag{8}$$

Speed of vehicles is set as 1 what means that the travel time between two clients is equal to the Euclidean distance between them. Visualization of the position of all customers located in the file that describes the RC201 test problem is presented in Fig. 2. The store is marked in red.

Experimental research relied on measurements and evaluation of influence each parameters CLONALG algorithm in terms of execution time and quality of the results.

For research purposes, six tests were selected from the Solomon's set, one from each group: C1, C2, R1, R2, RC1, RC2. Selected tests are: C106, C203, R101, R210, RC107 and RC204. Testing kits were selected in a random way in order to maximize their differing levels of difficulty.

A. The number of iterations

The number of iterations of the algorithm is equivalent to the number of generations of antibodies in the clonal selection algorithms. It is determined by the parameter N_{gen}.

The tests carried out showed that with increasing number of iterations, the quality of solutions is greater. At the same time, execution time is longer. Increasing the number of iterations, at the same time we increase the number of attempts to improve the solutions represented by antibodies. Any attempt to improve the solutions require additional operations that directly extends the execution time.

B. The size of pool

Size of the pool in the algorithm CLONALG is defined by a parameter N.

On the basis of experiments, increasing the size of the pool of antibodies N clearly improves the quality of results, while prolonging the performance time. The execution time increases

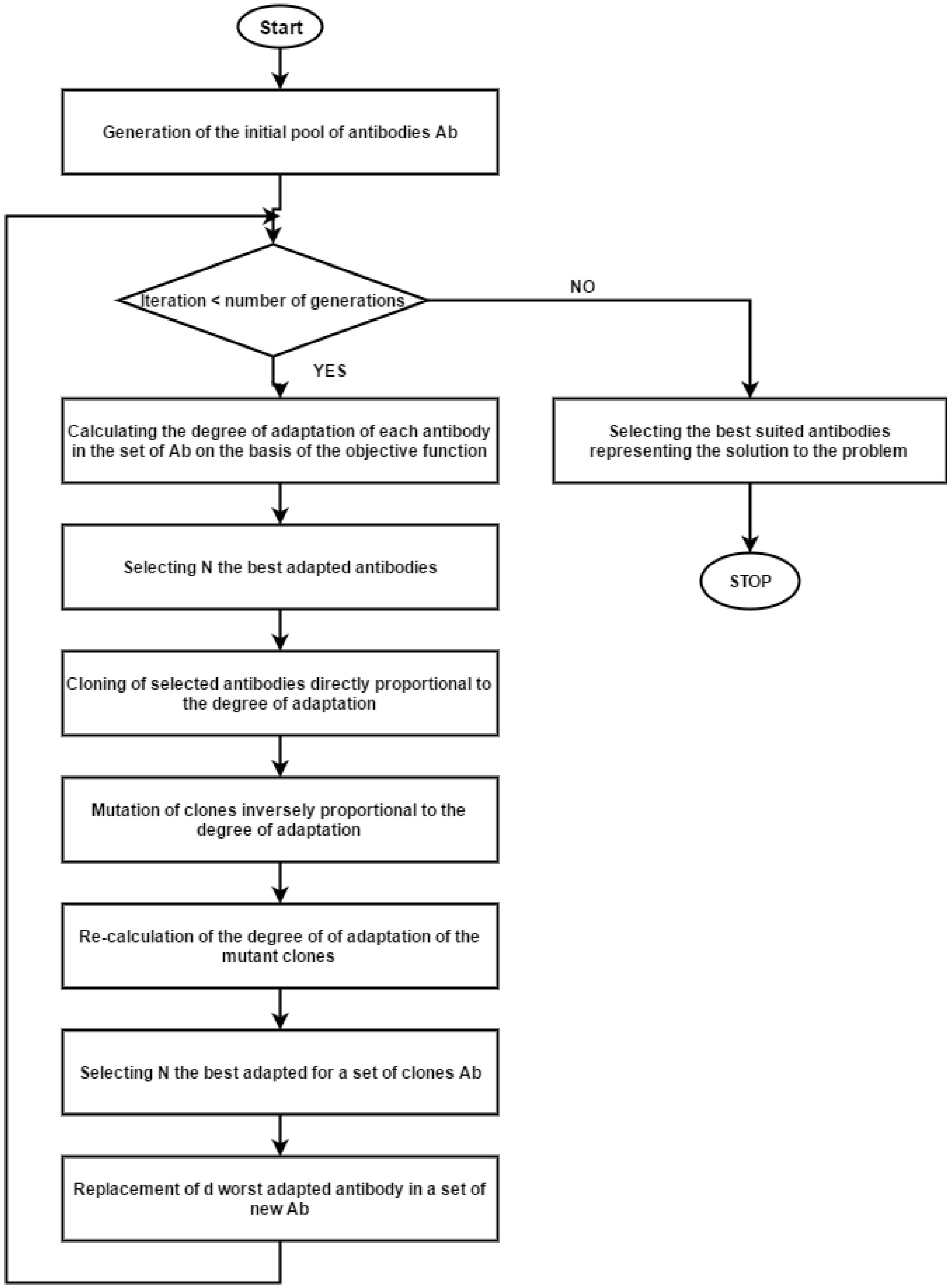

Fig. 1: The algorithm of the proposed method.

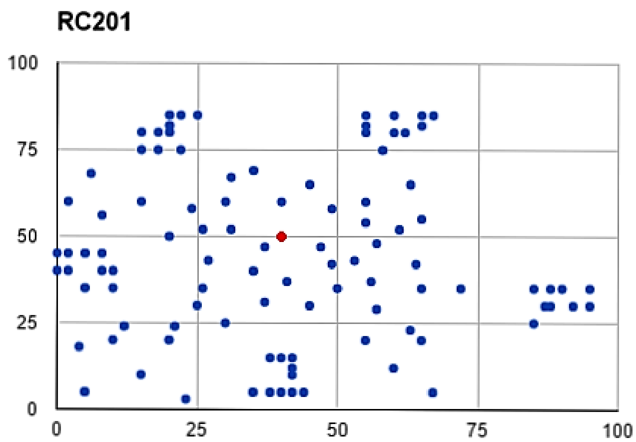

Fig. 2: Visualization of customers distribution in problem RC201

almost linearly. The impact of N is in line with expectations. Each antibody represents a solution to the problem. The more different antibodies is the pool, the greater the number of different possible solutions. This increases the probability that at least one of the available antibodies will be improved causing that it will be better than the best current solution.

C. The number of selected best-fit antibodies

The number of best-fit antibodies chosen for the cloning of the pool of all the antibodies determines the parameter n.

The parameter n has a smaller impact on the quality of obtained results than the parameters N_{gen} and N. In the case of increasing value of n, the execution time is higher. A smaller impact of n on the obtained results, can be explained by the fact that antibodies are selected in relation to the value of their adaptation. That is why the increasing of n causes selection of the worst antibodies.

D. The number of the least matched antibody replaced with new

The number of the least matched antibodies that will be overwritten is determined by the parameter d.

For the test configuration of algorithm, the parameter d had no significant effect on the results, as well the execution time of the algorithm. Replacing the least adapted antibodies on new allows to increase diversity in the pool of antibodies. In the case of function optimization, parameter allows to escape from local extreme in order to find global. At the same time, the probability of generating a new antibody with better adjustment than the current one is low. Therefore, setting the rate parameter d on $10 - 20\% * N$ seems to be a reasonable solution.

E. Coefficient of cloning

Coefficient of cloning antibodies is determined by the parameter β. It has a direct impact on the number of clones that will be created for each antibody subjected to cloning operation.

Coefficient of cloning β has a significant impact on the quality of results returned by the algorithm. At the same time, the increasing of β extends the execution time of the algorithm in a linear way. This happens because of β. larger the value of β. The higher is value, the more clones will be created from each of the antibodies. Thus, the probability is higher that at least one of the clones resulting from mutations will be improved. This will make that solution will be better than a result from which the clone was created.

F. The aspect of ratio distribution

The aspect of ratio distribution ρ directly affects the value of mutation α, which specifies the number of mutations that will be performed on the clone of antibody.

The results show that the aspect of ratio distribution ρ for the test configuration of the algorithm has an impact on the quality of the obtained solutions, but this effect is smaller than in the case of the parameters N_{gen}, N or β. Increasing the value of ρ, execution time is shorter. What is the consequence of the formula (5) – the higher is value of ρ, the fewer mutations is carried out on a clone of an antibody, which reduces execution time.

G. Analysis of the results

The conducted experimental research indicate that the greatest impact on the quality of the solutions returned by clonal selection algorithm implemented in the problem of vehicle routing with time windows have N_{gen}, N and β. The parameters n and ρ have much smaller impact than other parameters. The smaller has d. Moreover, parallel instructions have the greatest impact on the execution time of a given algorithm, but the impact is dependent on the number of available processors in the computer - the more threads, the shorter execution time is.

V. The obtained results

Minimizing the number of vehicles is the main criterion for optimization. In the 39 to 56 test problems, the algorithm equalized best known result in terms of vehicles number. In the remaining 17 problems, the number has always been only one more than the best known solution to solve that problem. The total length of the routes is an average of 10% longer. In addition, the total length of the route is shorter than the known solutions, although the number of vehicles is slightly higher in R103, RC101 and RC105.

To improve the obtained results, the clonal selection algorithm can be combined with local Search algorithms, thereby forming a hybrid algorithm. Another possibility is to use evolutionary strategies occurring in evolutionary algorithms.

Another possibility for improvement it is to use parallelism. The implemented algorithm uses multi-threading to improve execution time. The use of multi-threading in a manner similar to what has been applied in [20], [21] (in simulated annealing algorithm), in which the parallel running processes periodically send the best obtained solutions between each other. Such a solution should improve the accuracy of obtained solution.

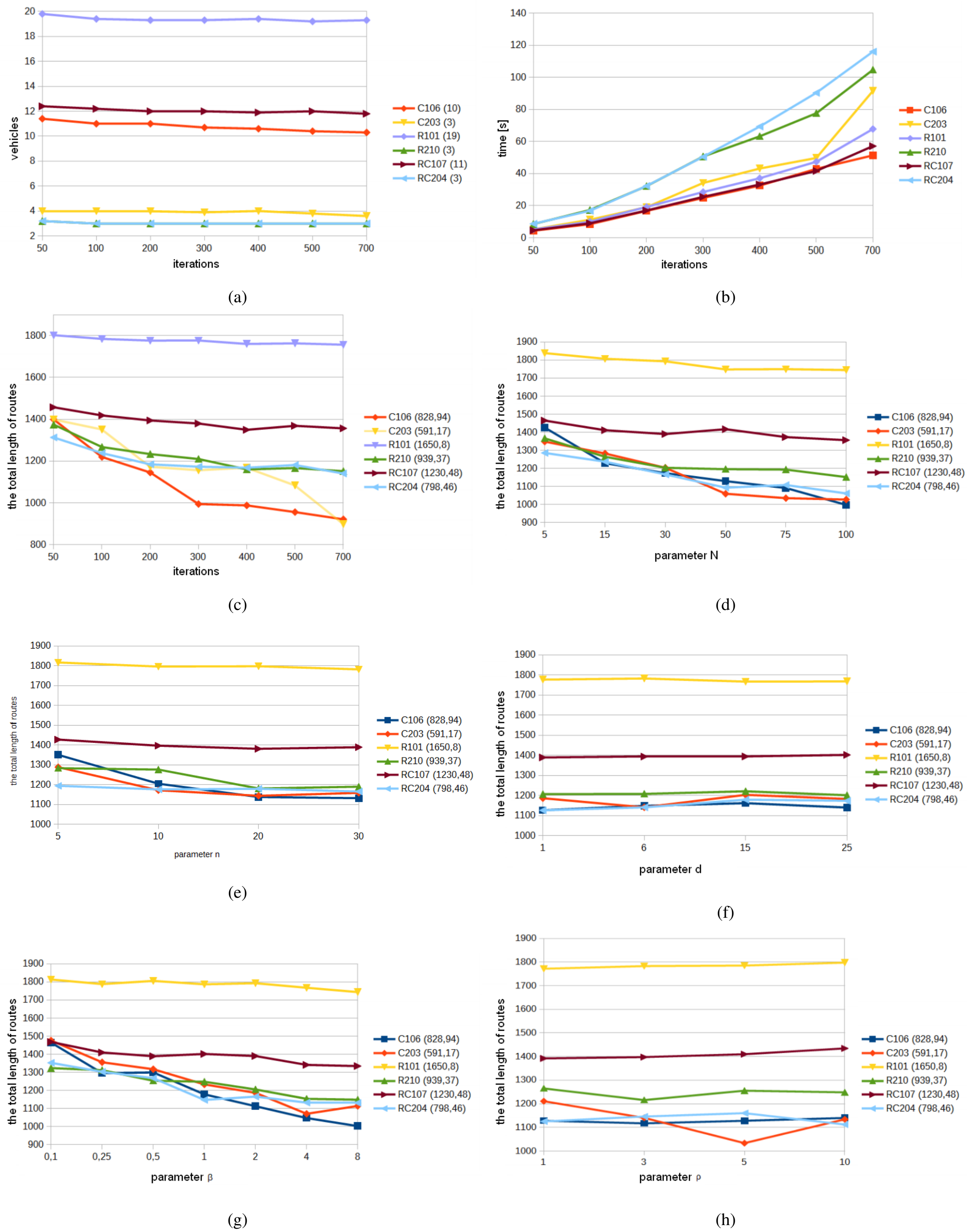

Fig. 3: Graphs of the obtained results for selected problems.

Another possibility is to make improvement in every thread, and then choose the best from among the obtained solutions. However, improvement in the quality of solutions extends time of the algorithm.

VI. Conclusions

In this paper, clonal selection algorithm called CLONALG was presented and used to solve the problem of vehicle routing problems with time windows (VRPTW). The effect of various parameters of the algorithm has been tested in terms of execution time and the quality of the results obtained in the described problem. The results can be considered as good, although they are different than the known solutions considered to be the best. However, in this study, the classical approach was analyzed without modification described in Sec. III. Further research on this topic may include modifications of the algorithm that would allow to obtain better solutions. It is a good idea to use other solutions, what is planned for future research.

References

[1] M. Woźniak, M. Gabryel, R. K. Nowicki, and B. A. Nowak, "An application of firefly algorithm to position traffic in nosql database systems," in *Knowledge, Information and Creativity Support Systems*. Springer, 2016, pp. 259–272.

[2] D. Połap, "Neuro-heuristic voice recognition," in *Computer Science and Information Systems (FedCSIS), 2016 Federated Conference on*. IEEE, 2016, pp. 487–490.

[3] M. Woźniak and D. Połap, "Voice recognition through the use of gabor transform and heuristic algorithm," *International Journal of Electronics and Telecommunications*, vol. 63, no. 2, pp. 159–164, 2017.

[4] D. Połap, M. Woźniak, C. Napoli, E. Tramontana, and R. Damaševičius, "Is the colony of ants able to recognize graphic objects?" in *International Conference on Information and Software Technologies*. Springer, 2015, pp. 376–387.

[5] M. Woźniak and Z. Marszałek, "An idea to apply firefly algorithm in 2d image key-points search," in *International Conference on Information and Software Technologies*. Springer, 2014, pp. 312–323.

[6] M. Woźniak, D. Połap, M. Gabryel, R. K. Nowicki, C. Napoli, and E. Tramontana, "Can we process 2d images using artificial bee colony?" in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2015, pp. 660–671.

[7] M. Woźniak, D. Połap, C. Napoli, and E. Tramontana, "Graphic object feature extraction system based on cuckoo search algorithm," *Expert Systems with Applications*, vol. 66, pp. 20–31, 2016.

[8] M. Wozniak, C. Napoli, E. Tramontana, G. Capizzi, G. L. Sciuto, R. K. Nowicki, and J. T. Starczewski, "A multiscale image compressor with rbfnn and discrete wavelet decomposition," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[9] D. Połap, "Designing mazes for 2d games by artificial ant colony algorithm," in *Proceedings of Symposium for Young Scientists in Technology, Engineering and Mathematics*, 2016, pp. 63–70.

[10] F. Bonanno, G. Capizzi, S. Coco, C. Napoli, A. Laudani, and G. L. Sciuto, "Optimal thicknesses determination in a multilayer structure to improve the spp efficiency for photovoltaic devices by an hybrid fem—cascade neural network based approach," in *Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2014 International Symposium on*. IEEE, 2014, pp. 355–362.

[11] F. Bonanno, G. Capizzi, A. Gagliano, and C. Napoli, "Optimal management of various renewable energy sources by a new forecasting method," in *Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2012 International Symposium on*. IEEE, 2012, pp. 934–940.

[12] C. Napoli, G. Pappalardo, E. Tramontana, and G. Zappalà, "A cloud-distributed gpu architecture for pattern identification in segmented detectors big-data surveys," *The Computer Journal*, vol. 59, no. 3, pp. 338–352, 2016.

[13] R. Damaševičius, "Optimization of svm parameters for recognition of regulatory dna sequences," *Top*, vol. 18, no. 2, pp. 339–353, 2010.

[14] P. Čisar, S. M. Čisar, and B. Markoski, "Implementation of immunological algorithms in solving optimization problems," *Acta Polytechnica Hungarica*, vol. 11, no. 4, 2014.

[15] J. A. White and S. M. Garrett, "Improved pattern recognition with artificial clonal selection?" in *International Conference on Artificial Immune Systems*. Springer, 2003, pp. 181–193.

[16] J. Brownlee, "Clonal selection theory & clonalg-the clonal selection classification algorithm (csca)," *Swinburne University of Technology*, 2005.

[17] I. Aydin, M. Karakose, and E. Akin, "A multi-objective artificial immune algorithm for parameter optimization in support vector machine," *Applied Soft Computing*, vol. 11, no. 1, pp. 120–129, 2011.

[18] B. Jason, "Clonal selection algorithms," *Technical Report 070209A*, 2007.

[19] F. J. V. Z. Leandro N. de Castro, "Learning and optimization using the clonal selection principle," vol. 6, no. 3, pp. 239–251, 2002.

[20] S. W. Mahfoud and D. E. Goldberg, "Parallel recombinative simulated annealing: a genetic algorithm," *Parallel computing*, vol. 21, no. 1, pp. 1–28, 1995.

[21] Z. J. Czech and P. Czarnas, "Parallel simulated annealing for the vehicle routing problem with time windows," in *Parallel, Distributed and Network-based Processing, 2002. Proceedings. 10th Euromicro Workshop on*. IEEE, 2002, pp. 376–383.