UDC 519.6

Titov P.Y., Sudakov A.V., Esther L.C., Hanlie S. Application of the ant colony method for orderly tracking of hyperparameters in models of pandemic disease development.

Titov P. Yurii
Moscow Aviation Institute (National Research University)
Sudakov A. Vladimir
Keldysh Institute of Applied Mathematics (Russian Academy of Sciences)
Esther Luna Colombini
Institute of Computing (IC) - University of Campinas, Av. Albert Einstein
Hanlie Smuts
University of Pretoria

Abstract. *The paper considers the possibility of using the ant colony method to select the optimal parameters for simulation models. To specify a set of parameters, a parametric graph is used, consisting of layers (a separate parameter) and vertices in a layer (a specific parameter value). For this graph, the route will be the choice of one vertex in each layer, i.e. one value for each parameter. Calculation of the optimal route in such a graph is carried out using the ant colony method. The paper proposes to use the ant colony method not for the convergence of the method to a certain solution, but for directed enumeration of various solutions. In this case, after the first finding of the optimal solution, the algorithm continues the search, since the convergence of the algorithm to a local extreme is possible. To take into account the considered solutions, the use of a hash table is proposed. The paper proposes modifications of the algorithm aimed at solving the problem of convergence of the algorithm to one solution and accelerating (on average) the number of solutions considered until the optimal one is obtained. To test the algorithm, the SIRVD model was considered, where the parameters were the coefficients of the differential equations (a problem of low dimension) and the initial state (a problem of high dimension). Optimization was carried out until the parameters of the SIRVD model were obtained, which describe the real data from the Our World in Data aggregator with the least error. In this paper, we studied the influence of the parameters of the ant colony method on the efficiency of the method.*

Keywords: *Ant colony method, optimization, simulation, CIR*

Introduction.

Pandemics in modern history are not uncommon. In 2019, the global health system was tested by the SARS-Covid-19 pandemic. The modern public has reacted with the introduction of a mask regime, the introduction of lock-down, the abrupt opening of quarantine zones and departments of medical centers. Considering the increased cargo and passenger traffic, restrictions bring great difficulties. For medical centers, it is expected that it will be possible to quickly (up to two weeks) create quarantine zones and open additional quarantine departments. These activities involve hiring new staff, purchasing supplies, training and organizing staff and residents (or patients). With this approach, situational management is often used, when the decision to open additional branches is given when the existing ones are filled.

To simplify management decision-making, it is necessary to predict the situation. Existing forecasting systems are usually based on statistical information and use machine learning methods. These systems require a sufficiently large training sample (i.e., the accuracy depends on the time of collection of statistical information and its accuracy).

To describe the process of changing the number of healthy, sick, recovered people, there are systems of differential equations, for example, SIR, SEIR, SIRVD, etc.

Where the number of people belonging to the categories is considered: Susceptible, Exposed, Infected, Recovered, Vaccinated, Dead. To tune the system, it is necessary to calculate the weight coefficients of the system of differential equations.

Simulation and analytical models can more accurately describe some of the processes taking place during a pandemic. These models are not based on statistical data, but on the description of the process of changing the state of the system over time. In such models, it is possible to take into account the methods of infection transmission, quarantine rules, various levels of quarantine and methods of testing the population, vaccination processes, transport, logistics processes, and others. The main problem of these systems is the need to check the adequacy of the model. Due to the large number of stochastic factors, a large number of accurate models are inadequate. For simulation and analytical models, the parameters can be varied and be both quantitative and qualitative. Calculation of model parameters, allowing building an adequate model, is a labor-intensive task. The use of expert assessments also requires a thorough check of the processes in the simulation model. But the availability of statistical data allows not only assessing the adequacy of the processes of the functioning of the model, but also the parameters of this model.

At present, it has become possible to transfer many optimization, computational tasks from humans to computers. The user of such a system describes sets of parameters, and the computing system selects the optimal parameters by enumerating them. Most often, enumeration of parameters is carried out by brute force methods, since the computer system solves not an optimization problem, but a calculation one.

The paper proposes to develop software that could rearrange parameter sets so that rational sets are calculated earlier than if they were considered sequentially. The input data of such a system are sets of parameters and, in the process of operation, the values of criteria for a particular set of parameters. Recent results such as [1-4] demonstrate that the problem of optimizing hyperparameters in large and multilayer models is a direct obstacle to scientific progress. There are similar systems, for example, the Bayesian optimizer (IBM Bayesian Optimization Accelerator (BOA)). In world practice, studies of the Bayesian optimizer are widespread [5-13]. IBM has taken an artificial intelligence approach based on Bayesian optimization, which builds and optimizes the model in real time to predict the most "promising" points that are calculated by existing tools. However, BOA-based solutions are expensive, require a separate computing cluster, and use metaheuristics to generate multiple hypotheses. These metaheuristics are commercially closed and are not subject to analysis. These features often become disadvantages of the Bayesian optimizer.

The paper considers the application of the developed modification of the ant colony method for directed enumeration of hyperparameters. The algorithm of the ant colony method, developed for finding the traveling salesman path [14-16], can be easily modified for parametric problems [15-21]. In the presented works, the task of the ant colony method is to find rational solutions, while the majority of ants (agents) must move along the same path. At the same time, modifications that allow ants to find new, non-repeating solutions at each iteration were not considered by the scientific community. This approach is necessary when the task.

Methods and techniques

The operation of the ant colony method requires a graph structure, along the arcs of which agents (ants) move. For parametric optimization, the graph familiar to the traveling salesman problem is, in fact, a set of linked lists [21-29]. Each list defines a set of parameter values and may be referred to as a layer. The algorithm selects one vertex in each layer (i.e. for each parameter), i.e. parameter value. To take into account the already considered vertices, it is supposed to enter the paths of agents in the Hash table. In general terms, the modified ant colony method for directed enumeration of hyperparameters is shown in Figure 1. The block "Calculation of criteria values for a certain set of parameters" is supposed to be performed based on the operation of a simulation or analytical model that takes a vector of parameter values as input, and, as output variables, gives the value of the optimality criteria for the set of parameters. This approach allows testing the operation of the modified ant colony method independently of a complex analytical or simulation model, for example, on a simple simulation model.

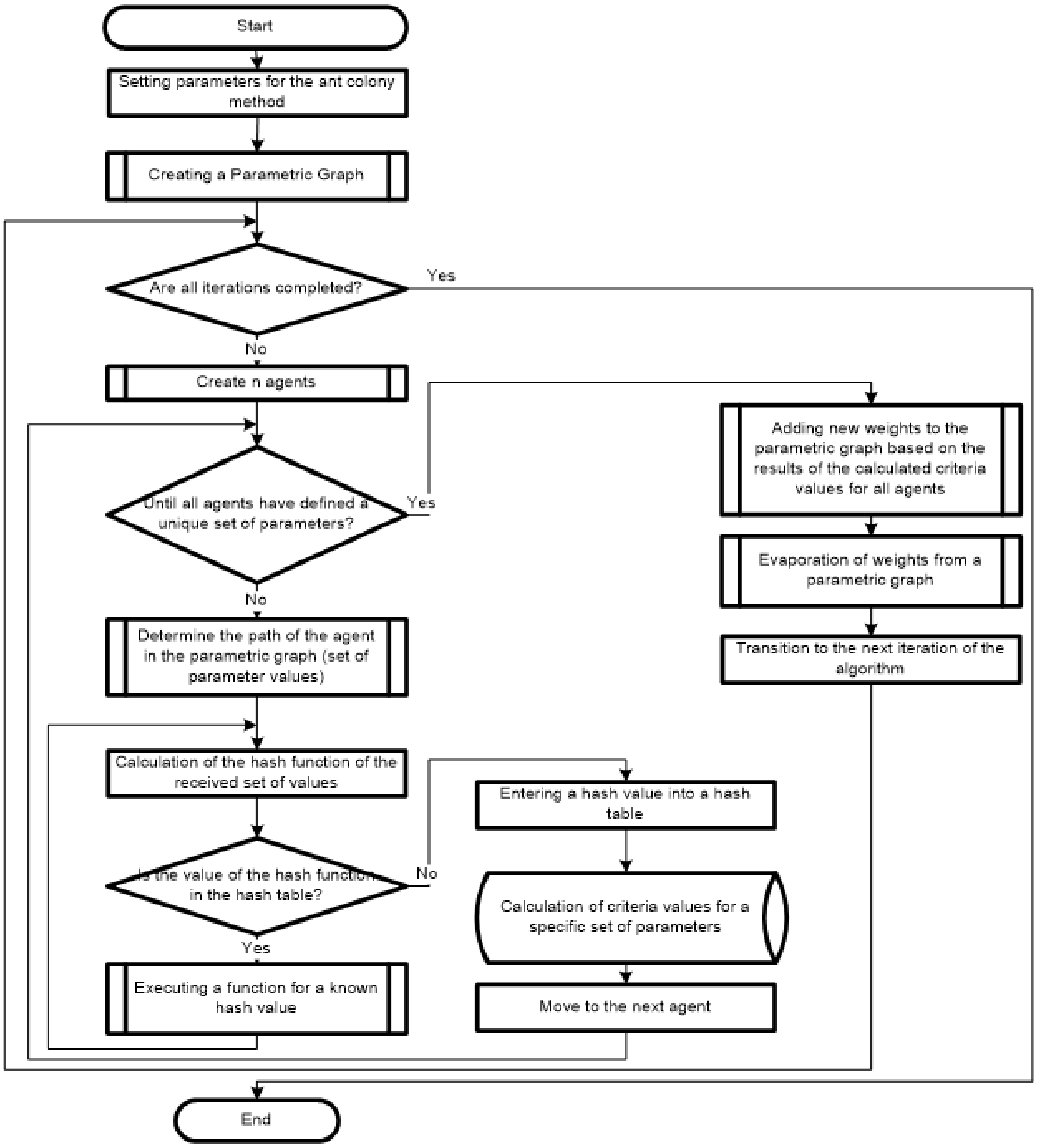
Start
Setting parameters for the ant colony method
Creating a Parametric Graph
Are all iterations completed?
Yes
No
Create n agents
Until all agents have defined a unique set of parameters?
Yes
No
Adding new weights to the parametric graph based on the results of the calculated criteria values for all agents
Evaporation of weights from a parametric graph
Transition to the next iteration of the algorithm
Determine the path of the agent in the parametric graph (set of parameter values)
Calculation of the hash function of the received set of values
Is the value of the hash function in the hash table?
No
Yes
Entering a hash value into a hash table
Calculation of criteria values for a specific set of parameters
Executing a function for a known hash value
Move to the next agent
End

Fig 1. Algorithm of the modified ant colony method.

The paper also proposes the use of two new modifications of the algorithm. The task of the original ant colony method is the convergence of all solutions to one, which interferes with the directed enumeration of hyperparameters. In addition, the absence of an alternative to the path length (for the traveling salesman problem), which allowed the initial weight (pheromone) to be entered on arcs with short lengths, negatively affects the performance of the algorithm. The algorithm without additional information converges to the solution that turned out to be the best at the first iterations.

As part of the modification of the algorithm, it is proposed for agents that have not found a new way in the graph to enter weights on the graph equal to 0, i.e. do not carry weight. Thus, you can try to reduce the impact of the fact that the algorithm tends to move all agents along the same route. The second modification is the need to roll back the state of the parametric graph to the initial state, if all agents have not found a single new solution at the iteration. Agents that have not found a new solution will be called "null" agents, and the iteration at which all agents are zero - "zero" iteration. The application of the proposed modifications is marked in black on the graphs. The algorithm without applying these modifications is gray lines.

To reduce the influence of weights and take into account the vertices visited by fewer agents, it is proposed to use the number of agents who visited the vertex during the entire time of algorithm (1) in the probabilistic formula for choosing the next transition.

$$P_{ij,k}(t) = \frac{k1 * {\tau^{k3}}_{norm,j}(t) + k2 * {{}^{1}\!/_{kol(t)_j}}^{k4}}{\sum\limits_{l \in J_{i,k}} (k1 * {\tau^{k3}}_{norm,j}(t) + k2 * {{}^{1}\!/_{kol(t)_j}}^{k4})}, j \in J_{i,k} \quad (1)$$

Where, $P_{ij,k}(t)$ – is the probability of choosing the j-th vertex from the i-th vertex by the k-th agent; $\tau_{norm,j}(t)$– is the normalized number of weights (pheromone) at vertex j; $kol(t)_j$ – is the number of visits to node j by agents. k1, k2, k3 and k4 are coefficients. The denominator in the formula is responsible for normalizing all values of the sums in such a way that the sum of all probabilities equals 1. All coefficients were taken equal to 1 during testing. The coefficient k2 varied. For a value equal to 0, the graphs were marked with a solid line, and for a value of 1, a dotted line.

Since the ant colony method is based on a probabilistic search for a vertex for the transition, it is necessary to conduct a lot of experiments and collect statistical information to analyze the operation of the algorithm. To study the possibility of applying the modification of the ant colony method for directed enumeration of hyperparameters, the following are estimated: the mathematical expectation of the solution number (serial number of the set of values) at which the optimal set of parameters was found, the mathematical expectation of the number of considered solutions, the frequency of finding the optimal solution for 10,000 iterations. For all estimates, confidence intervals are given at a significance level of 99%, if the interval itself on the graph does not merge with the calculated estimate of the mathematical expectation. The solutions and algorithms proposed in the work are implemented in the Python language as part of the MMK-Cluster software.

An analysis of the performance of the modified ant colony method is proposed to be carried out using the SIRVD model as an analytical model.

$$\frac{dS(t)}{dt} = -\frac{\beta * I(t) * S(T)}{N} + \sigma * R(t) - \alpha * S(t)$$

$$\frac{dI(t)}{dt} = \frac{\beta * I(t) * S(T)}{N} - \gamma * I(t) - \delta * I(t)$$

$$\frac{dR(t)}{dt} = \gamma * I(t) - \sigma * R(t)$$

$$\frac{dV(t)}{dt} = \alpha * S(t)$$

$$\frac{dD(t)}{dt} = \delta * I(t)$$

Experiment

For this model, based on historical data, it was necessary to calculate the coefficients $\beta, \sigma, \alpha, \gamma, \delta$. A parametric graph for finding the optimal set of coefficients is shown in Figure 2. It should be noted that at the moment the ant colony method only works with a discrete set of values, but there are studies and implementations for continuous optimization. For discretization, it is necessary to set a step, which determines the accuracy of the model and the size of the parametric graph. The parametric graph shown in Figure 2 contains 2.5 million solutions.

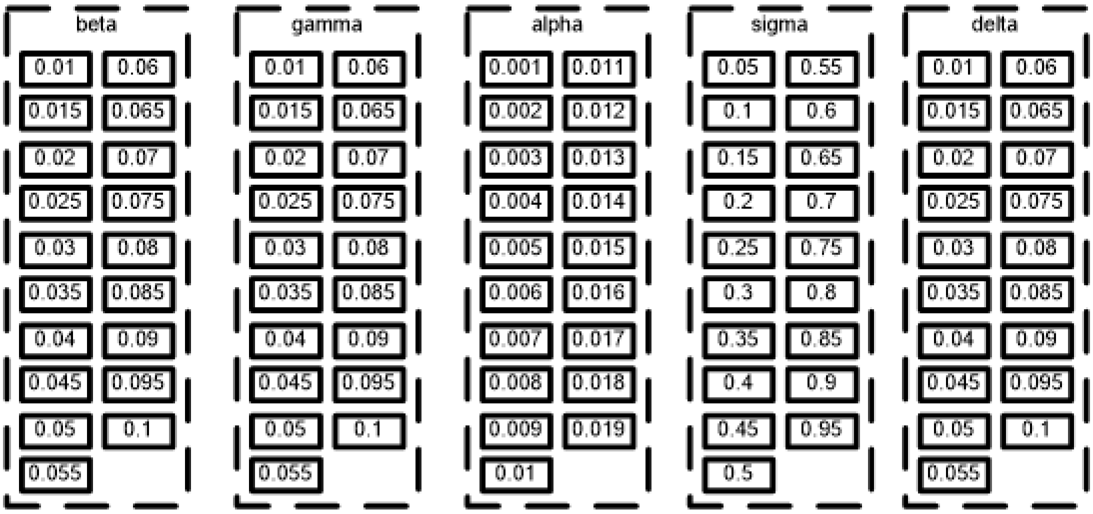

Fig. 2. An example of the parametric graph under study

The data for calculating parameter estimates were taken from the Our World in Data aggregator [30-31]. To determine the values of Infected (I), the total_cases parameter was used, for the number of Vaccinated (V) - total_vactination, for Mortality (D) - total_deths. Recovered and Susceptible values were determined based on the total population. It should be noted that the SIRVD model is not entirely accurate as only Susceptibles can become Infected. There are no transitions from Recovered (re-infected) and Vaccinated. Therefore, the data taken from the Our World in Data aggregator should not be well described by the SIRVD model. The algorithm for determining the value of the criterion was as follows:

1. For some initial state t, taking into account the parameters of differential equations obtained as a result of the ant colony method, the values were calculated $S(t+1), I(t+1), R(t+1), V(t+1), D(t+1)$
2. There is a comparison of the main indicators (the module of the number of infected, vaccinated and dead) calculated as a result of the algorithm with real indicators taken from the Our World in Data aggregator. Between iterations, the error accumulates.
3. Time increases by one step and becomes equal to t+1. The obtained values determine the initial state, and the transition to the 1st step is carried out until the data runs out.

Since it is clear from the work [32-33] that the estimates of the sought coefficients are not stationary, a small period of time with 10 measurements was subjected to the study. The initial state for the first experiment was statistical information from the aggregator on the first day of the study. The second experiment was carried out with the expansion of the decision graph and the addition of columns with the initial state of the system, i.e. Based on the available statistical data, the algorithm performed a retrospective analysis of the initial state. At the same time, for the second variant, the number of possible solutions sharply increases.

Results

The results of 100 runs of the algorithm of the modified ant colony method showed the possibility of using this method for directed enumeration of hyperparameters, in particular, to search for the coefficients of the SIRVD model. All modifications of the algorithm found the optimal set of coefficients of differential equations, considering less than 0.12% of possible solutions.

The convergence of the algorithm was studied by varying the “standard” parameters of the ant colony method: the number of agents per iteration (N), the evaporation coefficient (Ro) and the weight parameter (pheromone, Q). The most significant is the variation in the number of agents per iteration, since the number of considered solutions also depends on it.

The standard ant colony method, modified only to work with a parametric graph, loops on the best solutions found in the initial iterations. As a result, with the number of agents per iteration equal to 5, the optimal solution was found only in 25% of algorithm runs. For a graph of higher dimension (taking into account the definition of the initial state), this percentage drops to 10. The percentage of finding the optimal set of parameters by the standard algorithm approaches 100% only with 20 (35 for a graph of high dimension) agents at one iteration. This is due to the fact that a large number in the initial iterations allows us to consider more solutions at first. The use of the proposed modifications, both with zero agents and iterations, and k2=1, makes it possible to find optimal solutions in 100% runs even with the number of agents per iteration equal to 5.

The graphs (Fig. 3) show the results of collecting statistical information on the evaluation of the mathematical expectation of the solution number (data set number) based on the results of 100 runs. The thin dotted lines indicate the boundaries of the confidence intervals at a confidence level of 0.99. The upper graph corresponds to a small parametric graph, consisting of only one value of the coefficients of differential equations, and the lower graph corresponds to a large parametric graph.

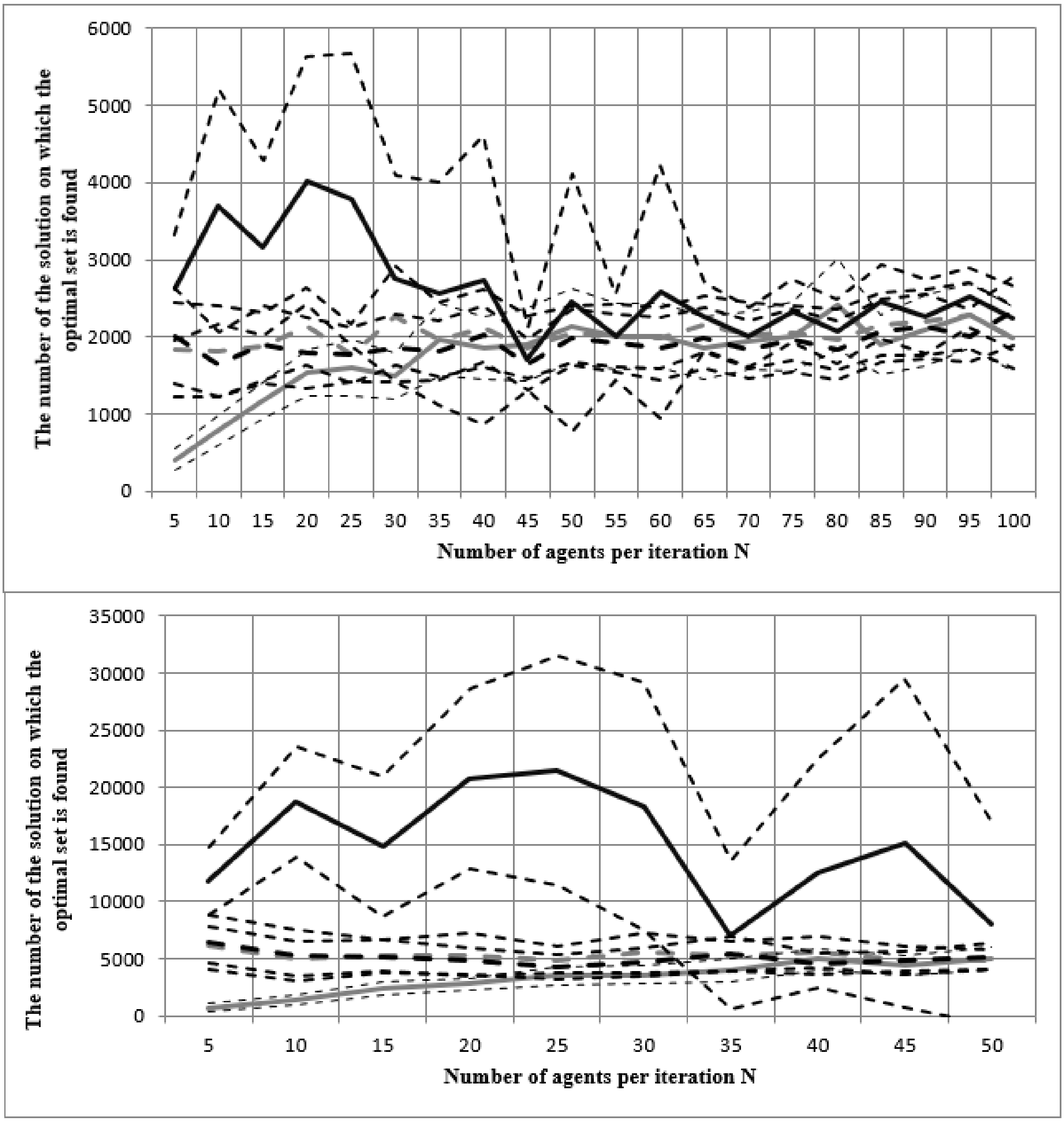

Fig. 3. Dependence of the estimate of the mathematical expectation of the number of the solution, on which the optimal set of parameters is found, on the number of agents per iteration.

It can be seen from the graphs that the low percentage of successful searches for solutions using the standard ant colony method (gray solid line) with the number of agents less than 20 is due to the fact that the optimal set of parameters must be found at the “first” iterations (the graph starts from the lowest values), otherwise the algorithm will go in cycles on the found non-optimal solution. Adding k2=1 (black solid graph) leads to finding the optimal set of parameters for any number of agents per iteration, but the solution itself was found after reviewing more than 0.2% (compared to 0.12%) of the solutions. With an increase in the number of agents per iteration, all algorithms converge to finding the optimal set of parameters when considering the same number of solutions (up to 3000 for a low-dimensional graph (2500000 options in total) and 5000 for a large one (56000000000 all options)). Algorithms taking into account zero agents and iterations for any number of agents find solutions as in the steady state. This property allows using a smaller number of agents per iteration, and thus reducing the running time of the algorithm.

Figure 4 shows the dependence of the estimate of the frequency of finding a new solution by an agent for various algorithms. This parameter shows how many solutions were considered by the algorithm for the allotted number of iterations. Finding by the agent a solution that is already in the Hash table does not lead to new results, and, therefore, this agent can be considered "fictitious".

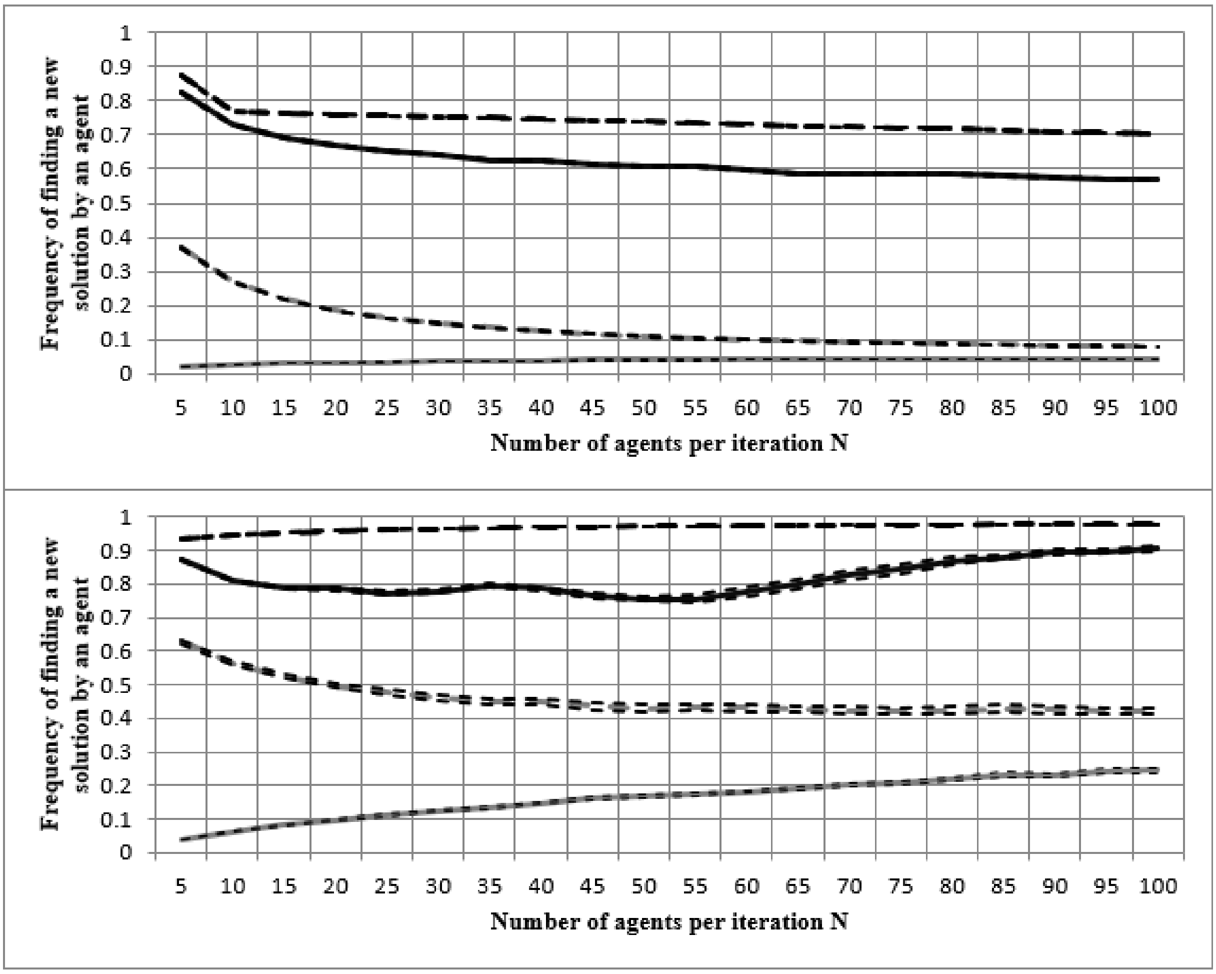

Fig. 4. Dependence of the estimate of the frequency of finding a new solution when searching for a path by an agent on the number of agents per iteration.

From the graphs in Figure 4, it can be seen that the standard method finds solutions that have already been considered. The probability of finding a new solution by the standard algorithm does not exceed 5% on a low-dimensional graph and tends to 40% on a high-dimensional graph. Adding actions for the case when a new solution is not found: special behavior conditions for zero agents and zero iterations (dotted lines), slightly correct the situation, but with an increase in the number of agents per iteration, they converge to the corresponding values. The best effect is achieved by setting the coefficient k2=1 (black lines). For a high-dimensional graph, the number of found unique solutions tends to 100%.

It should also be noted that the number of zero iterations, i.e. iterations in which no agent from the group found a new solution is very different for algorithms. So for a standard algorithm, on average, on a low-dimensional graph, 85% of iterations are zero, a little better for a high-dimensional graph (reaches 50%), when, as when taking into account zero agents and iterations, this indicator drops to 20% and 0.1%, respectively, and when applying k2=1 is equal to 0.

The Q parameter (the base value of the number of added weights) does not significantly affect the performance of the algorithm, since the relative rather than the absolute value of the weights is important for the probabilistic choice. For the evaporation parameter, the situation is similar to the studies obtained in other papers. It is recommended to set the evaporation parameter so that most of the weights in the parametric graph are based on the weights decreasing by a small factor.

Conclusion

The paper considers the possibility of using the ant colony method for directed enumeration of hyperparameters to solve the problem of optimizing model parameters. This study was carried out to consider an optimization algorithm in order to apply it to optimize the parameters of pandemic development models. In addition to considering the possibility of application, modifications of the algorithm are proposed that allow solving the problem of directed enumeration of hyperparameters.

To test the effectiveness of the algorithm, the problem of determining the coefficients of differential equations for the SIRVD model is considered. In addition to considering the coefficients of differential equations, the problem of determining the initial state was also considered. The assessment of the quality of the obtained models was determined on the basis of historical information obtained from the Our World in Data aggregator.

According to the results of the algorithm, the optimal set of parameters was found when considering less than 0.08% of the considered solutions, which makes it possible to guarantee the reduction of the required data set for directed enumeration of options. To take into account the considered sets of parameters, these sets are stored in a hash table. The paper proposes an algorithm for the ant colony method using a hash table. Suggested modifications of the algorithm:

- for agents that have not found a new solution (the solution is in the hash table), enter the weight on the graph in the amount of 0 (zero agents);
- if at the iteration no agent has found a new solution, then the parametric graph is transferred to the initial state (zero iteration)
- the ability to take into account how many agents have visited the vertex, and to give preference to those vertices visited by a smaller number of agents (taking into account standard weights (pheromone))

Based on the results of testing the algorithm, it is shown that the proposed modifications of the algorithm make it possible to carry out a directed enumeration of hyperparameters, while practically not finding identical solutions. When applying modifications, the number of the solution is stabilized, at which the optimal set of model parameters is found, with different parameters of the ant colony algorithm.

This algorithm with modifications is supposed to select parameters in order to find a set that ensures the maximum adequacy of the developed simulation model to real data. As a simulation model, it is planned to use the pandemic counteraction simulation model being developed, which takes into account not only the processes of infection and recovery of patients, but also contacts between people, the work of medical personnel, logistical and material support, staff training processes and social reactions of the population.

Despite the operation of the algorithm, there are difficulties during implementation:

- construction of a parametric graph at the moment is possible only when the space is discretized. But, since there are studies of the ant colony method, which works in a continuous space of variables, further improvement of the algorithm is expected.
- the algorithm has the curse of dimension and, in the case of a large number of vertices in a layer, does not optimize the problem well.
- at the moment, single-criteria optimization is considered, for which there are many alternatives, including those using machine learning. In the future, it is supposed to solve the problem of multiobjective optimization.

Acknowledgments

The reported study was funded by RFBR and CNPq, FASIE, DBT, DST, MOST, NSFC, SAMRC according to the research project № 20-51-80002

References

1. Nicolas Pinto, David Doukhan, James J. DiCarlo, and David D. Cox. A high-throughput screening approach to discovering good forms of biologically inspired visual representation. // PLoS Comput Biol, 5(11):e1000579, 11. 2009. https://doi.org/10.1371/journal.pcbi.1000579

2. Coates, A., Ng, A. &amp; Lee, H.. (2011). An Analysis of Single-Layer Networks in Unsupervised Feature Learning. // Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, in Proceedings of Machine Learning Research (Open Access 10.10.2022) https://proceedings.mlr.press/v15/coates11a.html.

3. A. Coates and A. Y. Ng. The importance of encoding versus training with sparse coding and vector quantization. // ICML'11: Proceedings of the 28th International Conference on International Conference on Machine Learning June 2011 Pages 921–928

4. James Bergstra, Remi Bardenet, Remi Bardenet, Balazs Kegl. Algorithms for Hyper-Parameter Optimization (Open Access 10.10.2022: https://proceedings.neurips.cc/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf)

5. Feurer, M., Hutter, F.. Hyperparameter Optimization. In: Hutter, F., Kotthoff, L., Vanschoren, J. (eds) Automated Machine Learning. // The Springer Series on Challenges in Machine Learning. Springer, Cham. 2019. https://doi.org/10.1007/978-3-030-05318-5_1

6. Koehrsen, Will. A conceptual explanation of bayesian hyperparameter optimization for machine learning. 2018. (Open Access 10.10.2022: https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f)

7. Bergstra, James S., Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. // In Advances in neural information processing systems, pp. 2546-2554. 2011.

8. Akiba, Takuya, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. // In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2623-2631. 2019. https://doi.org/10.48550/arXiv.1907.10902

9. https://krasserm.github.io/2018/03/21/bayesian-optimization (Open Access 10.10.2022)

10. https://krasserm.github.io/2018/03/19/gaussian-processes (Open Access 10.10.2022)

11. https://towardsdatascience.com/a-conceptual-explanation-of-bayesian-model-based-hyperparameter-optimization-for-machine-learning-b8172278050f (Open Access 24.08.2022)

12. Ian Dewancker, Michael McCourt, Scott Clark Bayesian Optimization Primer (Open Access 10.10.2022: https://static.sigopt.com/b/20a144d208ef255d3b981ce419667ec25d8412e2/static/pdf/SigOpt_Bayesian_Optimization_Primer.pdf)

13. IBM Bayesian Optimization Accelerator 1.1 helps identify optimal product designs faster with breakthrough performance for scientific discovery and high-performance computing simulation (Open Access 10.10.2022:

https://www.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/6/877/ENUSZP20-0186/index.html&request_locale=en)

14. Colorni A., Dorigo M., Maniezzo V. Distributed Optimization by Ant Colonies. // Proc. First Eur. Conf. on Artific. Life, Paris, France, F.Varela and P.Bourgine (Eds.), Elsevier Publishing. pp. 134-142, 1992

15. Dorigo, M., St¨ utzle, T.: Ant Colony Optimization //MIT Press, p. 321, 2004

16. Joseph M. Pasia, Richard F. Hartl, Karl F. Doerner. Solving a Bi-objective Flowshop Scheduling Problem by Pareto-Ant Colony Optimization M. Dorigo et al. (Eds.) // ANTS 2006, LNCS 4150, pp. 294–305, 2006

17. Torry Tufteland(B), Guro Ødesneltvedt(B), and Morten Goodwin Optimizing PolyACO Training with GPU-Based Parallelization M. Dorigo et al. (Eds.) // ANTS 2016, LNCS 9882, pp. 233–240, 2016. DOI: 10.1007/978-3-319-44427-7 20

18. Parpinelli, R., Lopes, H., Freitas, A.: Data mining with an ant colony optimization algorithm // IEEE Trans. Evol. Comput. 6(4), pp. 321–332, 2002

19. Bremer, Jörg and Sebastian Lehnhoff. “Constrained Scheduling of Step-Controlled Buffering Energy Resources with Ant Colony Optimization. // ANTS Conference, 2020.

20. Acevedo J., Maldonado S., Lafuente S., Gomez H., Gil P. Model Selection for Support Vector Machines Using Ant Colony Optimization in an Electronic Nose Application. In: Dorigo M., Gambardella L.M., Birattari M., Martinoli A., Poli R., Stützle T. (eds) // Ant Colony Optimization and Swarm Intelligence. ANTS 2006. Lecture Notes in Computer Science, vol 4150. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11839088_47

21. Sinicyn, I. N., Titov YU.P. Razvitie stohasticheskih algoritmov murav'inoj organizacii // Bionika – 60 let. Itogi i perspektivy. Sbornik statej Pervoj Mezhdunarodnoj nauchno-prakticheskoj konferencii, 17-19 dekabrya 2021 goda, g. Moskva / Pod red. A.P. Karpenko // M.: Associaciya tekhnicheskih universitetov. c. 210-220. 2022. DOI: 10.53677/9785919160496_210_220

22. Parpinelli, R., Lopes, H., Freitas, A.: Data mining with an ant colony optimization algorithm. // IEEE Trans. Evol. Comput. 6(4), pp 321–332, 2002/

23. Junior, I.C.: Data mining with ant colony algorithms. In: Huang, D.-S., Jo, K.-H., Zhou, Y.-Q., Han, K. (eds.) // ICIC 2013. LNCS, vol. 7996, pp. 30–38. Springer, Heidelberg, 2013

24. Martens, D., De Backer, M., Haesen, R., Vanthienen, J., Snoeck, M., Baesens, B.: Classification with ant colony optimization. // IEEE Trans. Evol. Comput. 11(5), pp 651–665, 2007

25. Hahulin G.f. Titov, YU. P. Sistema podderzhki reshenij postavok zapasnyh chastej letatel'nyh apparatov voennogo naznacheniya. // Izvestiya Samarskogo nauchnogo centra Rossijskoj akademii nauk. 2014. T. 16. № 1-5. s. 1619-1623.

26. Titov YU.P., Modifikacii metoda murav'inyh kolonij dlya resheniya zadach razrabotki aviacionnyh marshrutov. // Avtomatika i telemekhanika, 2015, vypusk 3. s. 108–124.

27. Titov, YU. P. Modifikacii metoda murav'inyh kolonij dlya razrabotki programmnogo obespecheniya resheniya zadach mnogokriterial'nogo upravleniya postavkami. // Sovremennye informacionnye tekhnologii i IT-obrazovanie. 2017. T. 13. № 2 s. 64-74. DOI 10.25559/SITITO.2017.2.222.

28. Sudakov V.A., Bat'kovskij A.M., Titov YU.P. Algoritmy uskoreniya raboty modifikacii metoda murav'inyh kolonij dlya poiska racional'nogo naznacheniya sotrudnikov na zadachi s nechetkim vremenem vypolneniya.// Sovremennye informacionnye tekhnologii i IT-obrazovanie. 2020. T. 16. № 2. s. 338-350. doi:10.25559/SITITO.16.202002.338-350

29. Sinicyn, I. N., Titov YU.P. Instrumental'noe programmnoe obespechenie analiza i sinteza stohasticheskih sistem vysokoj dostupnosti (XV) // Sistemy vysokoj dostupnosti. 2021. T. 17. № 4. s. 24-33. DOI 10.18127/j20729472-202104-02. – EDN YEGVMR.

30. Our World in Data. Coronavirus Pandemic (COVID-19). Open Access 10.10.2022 https://ourworldindata.org/coronavirus)

31. Data on COVID-19 (coronavirus) by Our World in Data. (Open Access 10.10.2022 https://github.com/owid/covid-19-data/tree/master/public/data/)

32. Zhifang Liao, Peng Lan, Xiaoping Fan, Benjamin Kelly, Aidan Innes, Zhining Liao, SIRVD-DL: A COVID-19 deep learning prediction model based on time-dependent SIRVD, Computers in Biology and Medicine, Volume 138, 2021, https://doi.org/10.1016/j.compbiomed.2021.104868.

33. Yuto Omae, Yohei Kakimoto, Makoto Sasaki, Jun Toyotani, Kazuyuki Hara, Yasuhiro Gon, Hirotaka Takahashi. SIRVVD model-based verification of the effect of first and second doses of COVID-19/SARS-CoV-2 vaccination in Japan[J]. Mathematical Biosciences and Engineering, 2022, 19(1): 1026-1040. doi: 10.3934/mbe.2022047