

Developing parallel real-coded genetic algorithms for decision-making systems of socio-ecological and economic planning

Andranik S. Akopov^{a,b}

E-mail: aakopov@hse.ru

Armen L. Beklaryan^{a,b}

E-mail: abeklaryan@hse.ru

Manoj Thakur^c

E-mail: manoj@iitmandi.ac.in

Bhisham Dev Verma^c

E-mail: bhishamdevverma@gmail.com

^aNational Research University Higher School of Economics

Address: 20, Myasnitskaya Street, Moscow 101000, Russia

^bCentral Economics and Mathematics Institute, Russian Academy of Sciences

Address: 47, Nakhimovky Prospect, Moscow 117418, Russia

^cIndian Institute of Technology Mandi

Address: Mandi, Himachal Pradesh 175005, India

Abstract

This article presents a new approach to designing decision-making systems for socio-economic and ecological planning using parallel real-coded genetic algorithms (RCGAs), aggregated with simulation models by objective functions. A feature of this approach is the use of special agent-processes, which are autonomous genetic algorithms (GAs) acting synchronously in parallel streams and exchanging periodically by the best potential decisions. This allows us to overcome the premature convergence problem in local extremums. In addition, it was shown that the combined use of different crossover and mutation operators significantly improves the time efficiency of RCGAs, as well as the quality of the decisions obtained (proximity to optimum), providing a more diverse population of potential decisions (individuals).

In this paper, several suggested crossover and mutation operators are used, in particular, a modified simulated binary crossover (MSBX) and scalable uniform mutation

operator (SUM), which is based on quantization of the feasible region of the search space (dividing the feasible region on small subranges with equal lengths) while taking into account the common amount of interacting agent-processes and the maximum number of internal iterations of GAs forming potential decisions through selection, crossover and mutation. Such a functional dependence of the parameters of heuristic operators on the corresponding process characteristics, aggregated with the combined probabilistic use of various crossover and mutation operators, makes it possible to get maximum effect from the multi-processes architecture. As a result, the computational possibilities of RCGAs for solving large-scale optimization problems (hundreds and thousands of decision variables, multiple objective functions) become dependent only on the physical characteristics of the existing computing clusters. This makes it possible to efficiently use supercomputer technologies.

An important advantage of the proposed system is the implemented integration between the developed parallel RCGA (implemented in C++ and MPI) and the simulation modelling system AnyLogic (Java) using JNI technology. Such an approach allows one to synthesize real world optimization problems in decision-making systems of socio-economic and ecological planning, using simulation methods supported by AnyLogic. The result is an effective solution to single-objective and multi-objective optimization tasks of large dimension, in which the objective functionals are the result of simulation modeling and cannot be obtained analytically.

Key words: real-coded genetic algorithms; multi-objective optimization; Pareto front; simulation modeling; AnyLogic.

Citation: Akopov A.S., Beklaryan A.L., Thakur M., Verma B.D. (2019) Developing parallel real-coded genetic algorithms for decision-making systems of socio-ecological and economic planning. *Business Informatics*, vol. 13, no 1, pp. 33–44

DOI: 10.17323/1998-0663.2019.1.33.44

Introduction

Currently, there is a need to design decision-making systems for socio-economic and environmental planning using simulation models aggregated with genetic optimization algorithms for solving large-scale optimization problems.

For the first time, a similar approach was proposed in [1–3] in which the developed multi-agent genetic optimization algorithm MAGAMO was presented. MAGAMO is aggregated through objective functions with a simulation model of a distance trading enterprise. In [1],

the objective functions, in particular, were the accumulated profit, the size of the active client base and the inventory turnover. At the same time, a similar model included five product categories, six cities and three customer segments, which, taking into account multiple restrictions and temporal granularity, characterized it as a large-scale optimization problem.

Note that MAGAMO [3] uses the dynamic interaction of synchronized intelligent agents, each of which is an autonomous genetic algorithm (GA) that implements an internal procedure for the formation of an archive of

non-Pareto solutions, for example, SPEA2 (Strength Pareto Evolutionary Algorithm) [4]. In MAGAMO, the dimensionality of the solved optimization problem is reduced by splitting the initial set of decision variables into small groups with their subsequent distribution between process agents (autonomous GAs) to minimize the size of local populations and the number of necessary (resource-intensive) recalculations of fitness function values, respectively. The MAGAMO algorithm was previously used, in particular, for the rational control of environmental modernization of enterprises that are stationary sources of harmful emissions [5], to determine the best geological and technical activities on wells [6], etc. In [7], some modification of this heuristic algorithm through inclusion of an adaptive mechanism provided improving values of GA parameters on the individual level of agent-processes depending on the optimization results (values of minimized target of functions, the rate of convergence, the hypervolume metric of the Pareto front, etc.).

At the same time, a significant deficiency of MAGAMO is the use of binary coding of decision variables values, which causes use of classical operators of a single-point and two-point crossover, as well as inversion (binary) mutation. As a result, the time-efficiency of the algorithm reduction if there is a need to search for solutions in a continuous space of high dimensionality, i.e. when wide values of feasible ranges are specified for decision variables (for example, $[-100, 100]$), and there are increased requirements for the precision of computations (when the number of bits of the mantissa is 2 or more).

Another problem is the weak mutual aggregation of agent-processes in MAGAMO and the need to synchronize their states (replication of the values of decision variables between processes) at each GA iteration, all of which significantly reduces the efficiency of process parallelization.

Therefore, it is necessary to create a fundamentally new parallel genetic algorithm using the mechanism of real coding, i.e. belonging to the class of RCGA algorithms (real-coded genetic algorithms) [8] and this is based on using new heuristic operators of the appropriate type providing a mechanism of periodic exchanges of the best potential decisions between agents-processes.

The purpose of this paper is to develop a multi-agent parallel real-coded genetic algorithm for solving multi-objective optimization problems (MA-RCGA-MO) aggregated through the objective functions with AnyLogic simulation models. In the result, there is provision for solving large-scale optimization problems in decision-making systems for socio-economic and environmental planning.

It should be noted that the choice of the AnyLogic system is mainly prompted by the important advantages of the platform, such as supporting system dynamics methods, discrete event modeling and agent-based modeling within one model [9]. This allows you to design decision-making systems that require the development and use of complex simulation models. Examples are the system of rational control of ecological-economic systems [10–13], the system of modelling and optimization of the set of investments of a vertically integrated oil company [14–15], the system of optimal distribution of flows of requests for loans at the inter-regional underwriting center of a very large bank [16], the system of control of intellectual agent-rescuers behavior in the simulation of human crowd behavior in an emergency [17–18] and other systems.

1. Multi-agent parallel real coded genetic algorithm

Currently there is a line of well-known research on genetic optimization algorithms designed to solve multi-objective optimization problems. Among the most often used methods, the following algorithms should be

highlighted: SPEA2 [4], MOEA/D (multi-purpose GA based on decomposition) [19], NSGA-II and NSGA-III (multi-objective GA based on non-dominated sorting) [20, 21] and some other algorithms. In addition, the application of agent-based modelling for implementing GA is known as well. In particular, the MAGA algorithm [22] that is intended for solving large-scale optimization problems should be noted. Despite the multiple advantages of developed GAs, most of them use the binary coding mechanism for decision variables, which causes significant loss of time-efficiency when searching for solutions in a large-scale continuous space. Accordingly, this limits the possibility of using such GAs in designing decision-making systems based on simulation modelling of the behavior of complex objects. In order to overcome these difficulties, a new multi-agent parallel real-coded genetic algorithm is proposed. The algorithm is intended for solving large-scale multi-objective optimization problems.

The main features of the suggested algorithm are the following:

- ◆ using well-known crossover and mutation operators designed for real-coded genetic algorithms (RCGAs), such as SBX crossover (simulated binary crossover) [23], Laplace crossover (LX) [24], power mutations (PM) [25] and others;

- ◆ using new (modified) heuristic crossover and mutation operators, the characteristics of which functionally depend on the individual number of the associated agent-process (i.e., the process in which they are performed). This makes it possible to significantly improve their efficiency, in particular, to achieve better diversity of potential decisions, to provide splitting (quantization) of search ranges into a larger number of short intervals and, thus, to use maximally the capabilities of a multi-cluster (multiprocessor) computing system while increasing the time-efficiency of GAs;

- ◆ combined use of various heuristic opera-

tors (both existing and proposed) at the individual level of interacting agents-processes for the formation of new potential decisions (offspring-individuals);

- ◆ adding internal iterations to the GA, providing the generation of a larger number of offspring-individuals and potential decisions, respectively;

- ◆ providing the mechanism for periodically exchanging the best potential decisions between agents-processes to avoid the jamming problem of the GA at local extremes and achieve an acceptable rate of population evolution for large-scale optimization problems.

An abstract description of the multi-agent parallel genetic algorithm so developed is given below.

Here,

$i = 1, 2, \dots, n$ – the index of decision variables defining the values of the objective functions;

$\{p_{i1}, p_{i2}\}$ – the pair of parent decision variables (parent-individuals) formed in the result of the selection procedure (for example, using tournament selection) for all i -ths decision variables ($i = 1, 2, \dots, n$);

$\{\hat{x}_{i1}, \hat{x}_{i2}\}$ – the pair of descendants (offspring-individuals) formed by parents for all i -ths decision variables ($i = 1, 2, \dots, n$);

$u(a, b), l(a, b), s(a, b)$ – random numbers evenly distributed on the range of $[a, b]$;

$(k = 1, 2, \dots, K)$ – the index of parallel agent-processes (GA), where K is the maximum number of agent-processes in parallel GA;

$g_k = 1, 2, \dots, G_k$ – the index of internal iterations belonging to the k -th agent-process.

The following new heuristic operators are suggested for the real-coded genetic algorithm:

- ◆ **modified simulated binary crossover (MSBX)**, provided the generation of potential decisions in the continuous search space:

$$\eta = 2^{-1} \left(2^{-u(0,1)} \right)^{N-1}, \quad \kappa = \eta - 1, \quad (1)$$

$$\hat{x}_{i1} = \eta p_{i1} + \kappa p_{i2}, \quad (2)$$

$$\hat{x}_{i2} = \kappa p_{i1} + \eta p_{i2}, \quad (3)$$

$$p_{i1}, p_{i2} \in [a_i, b_i], p_{i1} \neq p_{i2}, \quad (4)$$

$$N = g_k + 2, \quad (5)$$

$$i = 1, 2, \dots, n, k = 1, 2, \dots, K, g_k = 1, 2, \dots, G_k,$$

where η, κ are coefficients (parameters of a crossover), N is the parameter simulated the number of bits in GAs with a binary coding ($N \in [2, G_k]$);

◆ **modified discrete SBX-crossover (DMSBX)**, provided the generation of potential decisions in the discrete search space:

$$\hat{x}_{i1} = \lfloor \eta p_{i1} + \kappa p_{i2} + 0,5 \rfloor, \quad (6)$$

$$\hat{x}_{i2} = \lfloor \kappa p_{i1} + \eta p_{i2} + 0,5 \rfloor, \quad (7)$$

$$p_{i1}, p_{i2} \in [a_i, b_i], p_{i1} \neq p_{i2}, \quad (8)$$

$$i = 1, 2, \dots, n.$$

◆ **scalable uniform mutation operator (SUM)**, provided quantizing of the feasible ranges of decision variables into uniform intervals to obtain potential solutions outside the area of local extremes:

$$\tilde{a}_i = a_i + \frac{(b_i - a_i)}{G_k \times K} g \cdot k, \quad \tilde{b}_i = \tilde{a}_i + \frac{(b_i - a_i)}{G_k \times K}, \quad (9)$$

$$\hat{x}_{i1} = l(\tilde{a}_i, \tilde{b}_i), \quad \hat{x}_{i2} = s(\tilde{a}_i, \tilde{b}_i), \quad (10)$$

$$i = 1, 2, \dots, n, g_k = 1, 2, \dots, G_k, k = 1, 2, \dots, K.$$

Note that all considered heuristic operators are executed with a given probability. At the same time, the probability of the execution of a crossover operator at each iteration of the GA is close to one (that is, the crossover is the most important GA operator with real coding). The probability of a mutation operator is selected taking into account the relief of the objective functions of the solved problem and, as a rule, it is at the range [0.001, 0.1] while minimizing the objective functions having relatively simple

relief, and in the range [0.1, 0.5] while minimizing complex objective functions with multiple local extremes located near the global optimum.

Here,

$t_k = 1, 2, \dots, T_k$ – index of the external iterations of the k -th agent process of GA, where T_k is the number of external iterations;

$g_k = 1, 2, \dots, G_k$ – the index of internal iterations of the k -th agent process (GA), where G_k is the number of internal iterations;

{*LX, SBX, MSBX, DMSBX*} – the set of possible crossover operators chosen with equal probability at each t_k -th step of GA, where *LX* is the Laplas crossover, *SBX* – the standard SBX-crossover, *MSBX* – the modified SBX-crossover, *DMSBX* – the modified discrete SBX-crossover;

{*PM, UM, DUM, SUM*} – the set of possible mutation operators chosen with equal probability at each t_k -th step of GA, where *PM* is the power mutation operator, *UM* – the standard operator of a uniform mutation, *DUM* – discrete operator of a uniform mutation, *SUM* – a scalable operator of a uniform mutation;

ω – the frequency of exchanging the best potential decisions between all k -th process agents ($k = 1, 2, \dots, K$).

Thus, the aggregated block diagram of the proposed multi-agent parallel real-coded GA developed for multi-objective optimization (MA-RCGA-MO) can be presented in the following form (*Figure 1*).

Note that the proposed GA is implemented for the each parallel agent-process that periodically exchanges the best (non-dominant Pareto) potential decisions through the global archive with all other agent- processes. Such an approach can significantly increase the rate of searching the Pareto-optimal solutions and overcome the problem of a premature convergence associated with frequent jamming of GA at local extremes. *Figure 2* shows the aggregated architecture of the developed decision-making system in which the AnyLogic

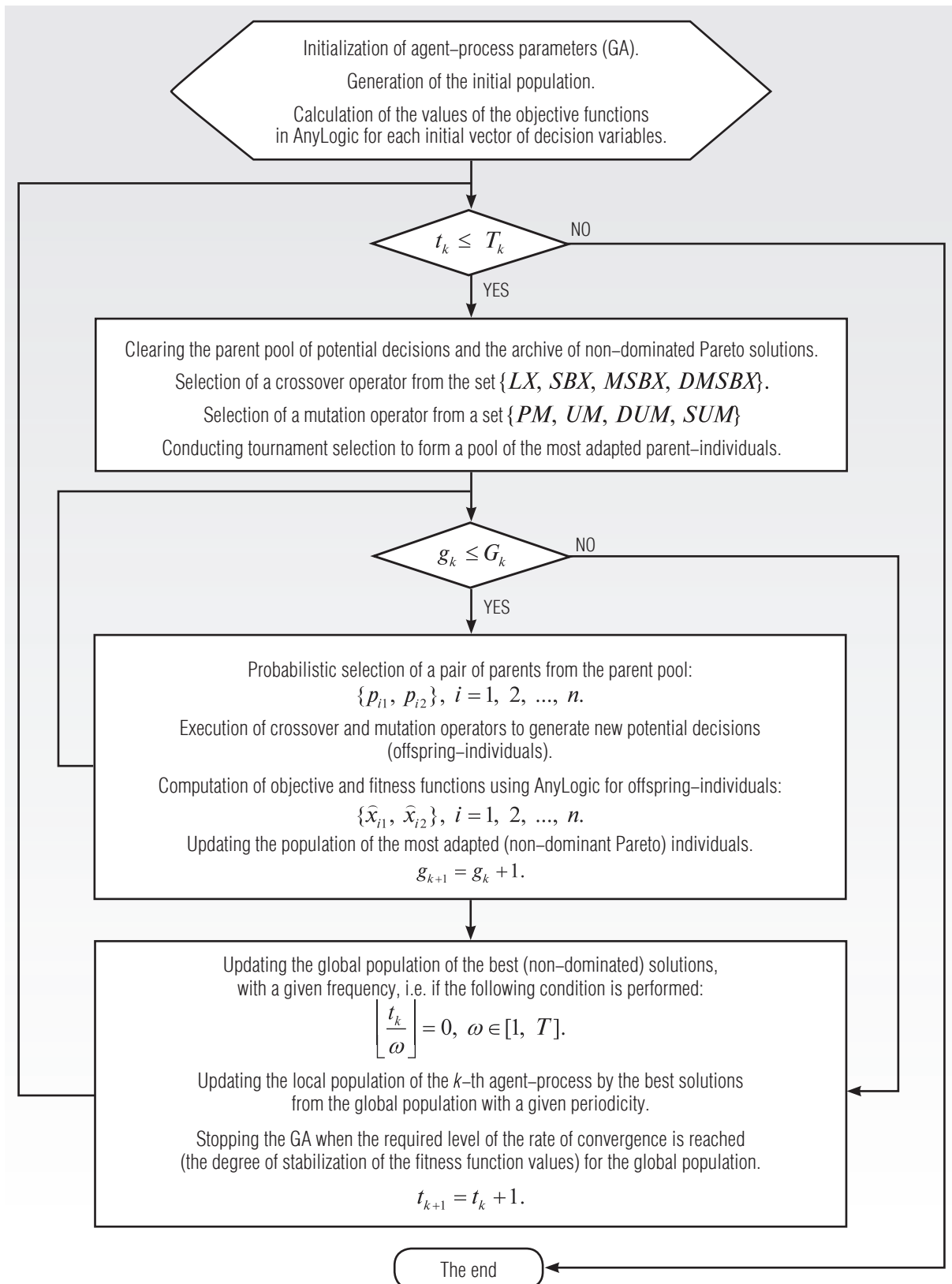


Fig. 1. The block diagram of the developed multi-agent parallel real coded GA

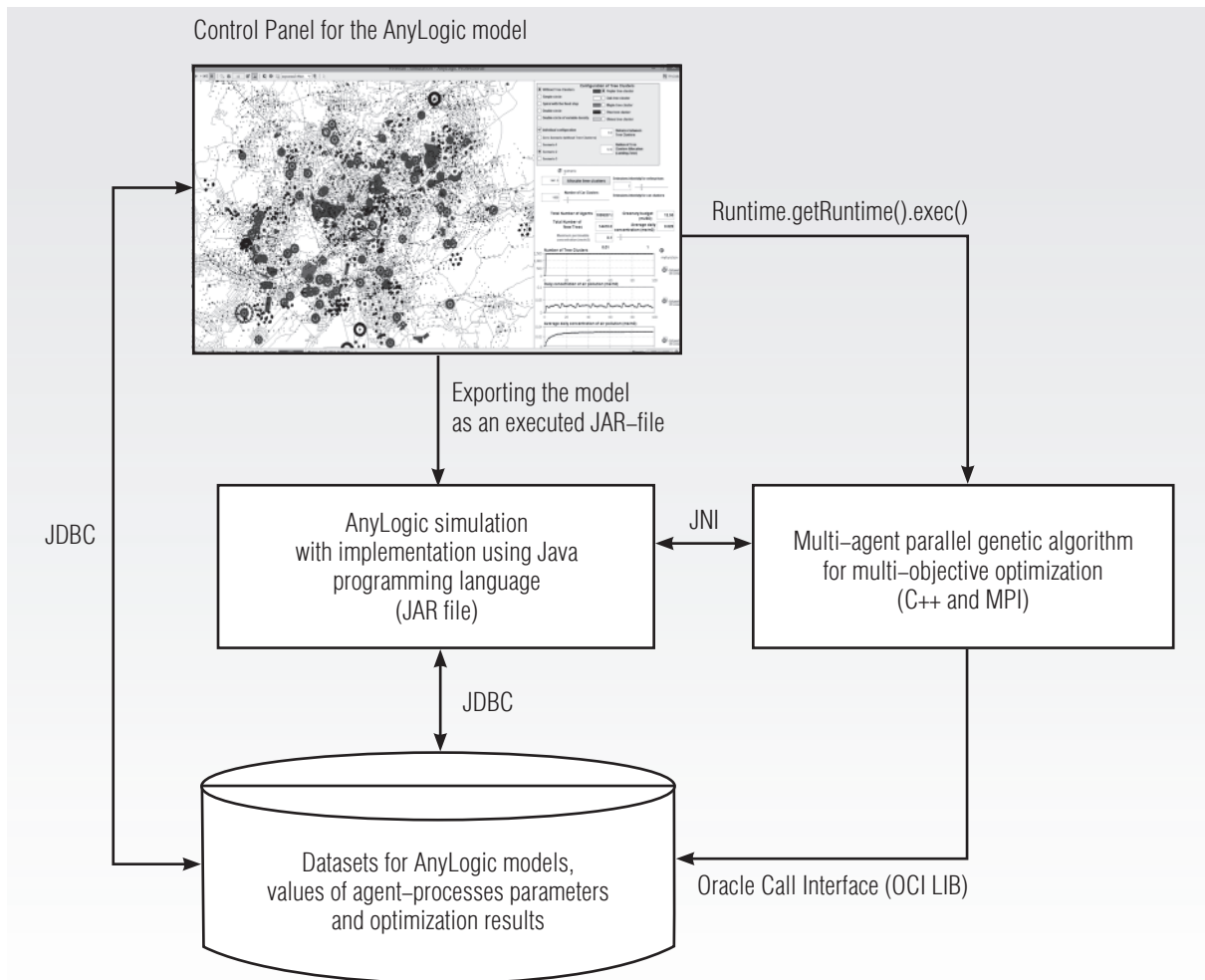


Fig. 2. The aggregated architecture of the decision-making system for socio-economic and ecological planning

simulation models are integrated with the developed genetic algorithm (MA-RCGA-MO). The algorithm is implemented in C++ programming language using the MPI (Message Passing Interface) technology, which makes it possible to provide an efficient procedure of data exchange between all agents-processes. Different software tools (for example, GIS maps, graphs, tables, etc.) can be used to provide the presentation of optimization results previously written in the database of the system (Oracle).

¹ <http://www.mpjexpress.org/>

An important advantage of the suggested architecture is the integration of the developed parallel GA with AnyLogic simulation models (implemented on Java) using the JNI technology (Java Native Interface). Note that currently there are parallelization technologies for the Java platform, for example, MPJ¹, which can also be applied to the AnyLogic models. However, when solving large-scale optimization problems, the most important factor is the performance of the corresponding computational procedures that can be significantly improved only by using C++ and MPI technologies.

2. An example of the practical implementation of the system for support of decision-making

An example of a decision-making system implemented for ecological and economic planning tasks related to the rational greening of the city is considered further. Earlier we developed an agent-based simulation model of the distribution of harmful emissions in the city (in the AnyLogic system) on the example of Yerevan, Republic of Armenia [10]. Initially searching for the best solutions in the model was conducted using a parallel genetic algorithm with binary coding, which required a lot of time for the generation of a subset of the Pareto optimal solutions (several hours of iterative calculations) on the server HP ProLiant DL 380 GB with two 6 core processors Intel Xeon CPU E5645, 2.4 GHz and 64GB of RAM, due to the large dimensionality of the optimization problem being solved.

It should be noted that in the simulation model two minimized objective functions were defined. The first is the average daily pollution concentration estimated in protected urban areas (in particular, in the areas of kindergartens), as well as the budget needed for greening the city to ensure the natural protection of socially important objects from harmful emissions produced by enterprises and transport. At the same time, 111 kindergartens were previously selected for protection by trees at the individual level, taking into account the variability of such parameters as the type of trees (for example, poplar, maple, oak, spruce, elm), the distance between the clusters of trees (from 5 to 60 meters), the radius of the planting zone (from 30 to 100 meters) and the geometry of planting trees around kindergartens (for example, a simple circle, an arithmetic spiral, a double circle, etc.).

In the result of the application of the developed multi-agent parallel real-coded genetic algorithm (MA-RCGA-MO), the time-efficiency of the search procedure for optimum solutions to the considered problem of complex ecological and economic system of the city was

significantly improved. Using MA-RCGA-MO, the best scenario was found for a polynomial time, providing almost fourfold reduction in the concentration of harmful emissions in the atmosphere in protected urban areas with an acceptable level of greening expenses. The optimization results, previously saved on the Oracle DBMS, were visualized on the map of Yerevan using the AnyLogic system (*Figure 3*).

Conclusion

This paper presents a new multi-agent parallel real-coded genetic algorithm MA-RCGA-MO, which provides an effective procedure for finding Pareto optimal solutions in large-scale multi-objective optimization problems.

An important feature of the suggested genetic algorithm is use of new heuristic crossover and mutation operators, the characteristics of which functionally depend on the number of the associated agent-process, as well as providing a mechanism for periodic exchange of the best potential decisions between all agents-processes to avoid a premature convergence (caused by potential jamming the GA at local extremes) and increase the rate of search for optimal solutions.

The important advantage of the suggested multi-agent GA is its aggregation with the AnyLogic simulation models through objective functions. At the same time, C++ programming language and MPI technology provide an effective procedure for periodically exchanging the best potential decisions, and the JNI technology provides the ability to integrate the GA with the AnyLogic models.

In future work, it is planned to implement different approaches to the generation of Pareto optimal solutions (for example, NSGA-III) for the developed multi-agent parallel genetic algorithm with studies of the effectiveness of appropriate modifications. Moreover, we expect the implementation of multi-agent genetic algorithm using technology CUDA (Compute Unified Device Architecture). ■

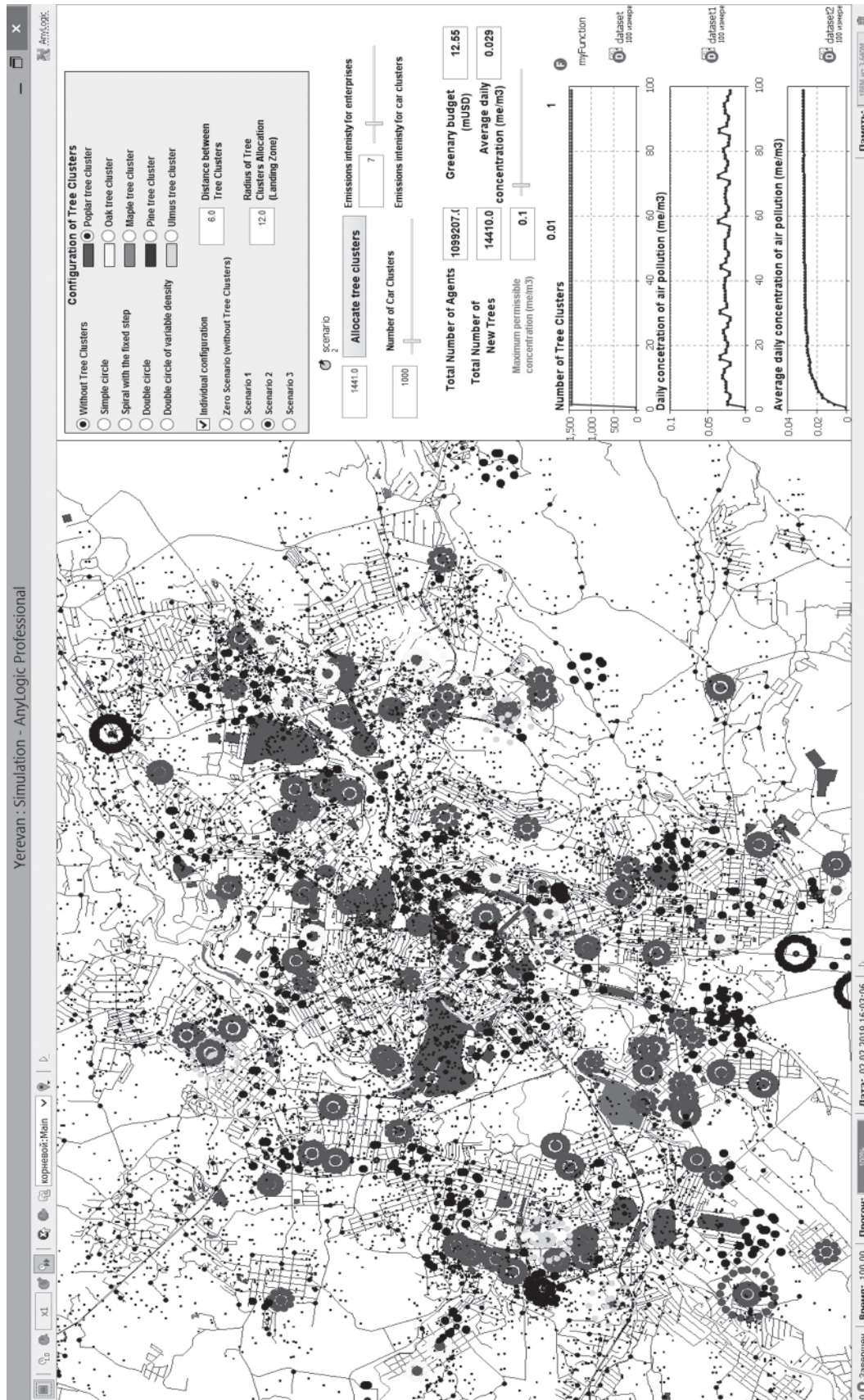


Fig. 3. Visualization of the results of minimization of harmful emissions in the city using the proposed genetic algorithm

Acknowledgements

This work was supported by the Russian Foundation for Basic Research (grant No. 18-51-

45001), as well as the Department of Science and Technology, Government of India (project No. INT/RUS/RFBR/305).

References

1. Khivintcev M.A., Akopov A.S. (2014) Application of multi-agent genetic algorithm for search of optimum strategic and operational decisions. *Business Informatics*, no 1 (27), pp. 23–33 (in Russian).
2. Khivintcev M.A., Akopov A.S. (2013) Distributed evolutionary network for the solution of multi-objective optimizing problems in simulation systems. *Business Informatics*, no 3 (25), pp. 34–40 (in Russian).
3. Akopov A.S., Hevencev M.A. (2013) A multi-agent genetic algorithm for multi-objective optimization. Proceedings of the *2013 IEEE International Conference on Systems, Man and Cybernetics. Manchester, UK, 13–16 October 2013*, pp. 1391–1395.
4. Bleuer S., Brack M., Thiele L., Zitzler E. (2001) Multiobjective genetic programming: Reducing bloat by using SPEA 2. Proceedings of the *2001 Congress on Evolutionary Computation (CES–2001). Seoul, South Korea, 27–30 May 2001*, vol. 1, pp. 536–543.
5. Akopov A.S., Beklaryan L.A., Saghatlyan A.K. (2017) Agent-based modelling for ecological economics: A case study of the Republic of Armenia. *Ecological Modelling*, vol. 346, pp. 99–118.
6. Akopov A.S., Beklaryan A.L., Khachatryan N.K., Fomin A.V. (2017) The forecasting system dynamics of oil production using simulation modeling. *Information Technologies*, vol. 23, no 6, pp. 431–436 (in Russian).
7. Akopov A.S., Beklaryan A.L., Khachatryan N.K., Fomin A.V. (2018) Development of an adaptive genetic optimization algorithm using agent modeling methods]. *Information Technologies*, vol. 24, no 5, pp. 321–329 (in Russian).
8. Herrera F., Lozano M., Verdegay J.L. (1998) Tackling real-coded genetic algorithms: operators and tools for the behaviour analysis. *Artificial Intelligence Review*, vol. 12, no 4, pp. 265–319.
9. Akopov A.S. (2014) *Simulation modeling*. Moscow: Urait (in Russian).
10. Akopov A.S., Beklaryan G.L., Beklaryan L.A. (2017) Agent-based modelling for ecological-economics system of a city (on example of Yerevan, Republic of Armenia). *Artificial Societies*, vol. 12, no 3-4. Available at: <https://artsoc.jes.su/s207751800000109-8-1> (accessed 01 September 2018) (in Russian).
11. Akopov A.S., Beklaryan A.L., Beklaryan L.A., Saghatlyan A.K. (2016) Modelirovanie regional'noj ehkologo-ehkonomicheskoy sistemy s mekhanizmom gosudarstvennogo regulirovaniya na primere Respubliki Armeniya [Modelling the regional ecological-economic system with the mechanism of the government regulation for the case study of the Republic of Armenia]. *Economics of Contemporary Russia*, vol. 72, no 1, pp. 109–119 (in Russian).

12. Beklaryan L.A., Akopov A.S., Beklaryan A.L., Saghatelyan A.K. (2016) Agent-based simulation modelling for regional ecological-economic systems. A case study of the Republic of Armenia. *Journal of Machine Learning and Data Analysis*, vol. 2, no 1, pp. 104–115.
13. Akopov A.S., Beklaryan L.A., Beklaryan A.L., Saghatelyan A.K. (2014) The integrated model of eco-economic system on the example of the Republic of Armenia. *Computer Research and Modeling*, vol. 6, no 4, pp. 621–631 (in Russian).
14. Akopov A.S. (2014) Parallel genetic algorithm with fading selection. *International Journal of Computer Applications in Technology*, vol. 49, no 3/4, pp. 325–331.
15. Akopov A.S. (2012) Designing of integrated system-dynamics models for an oil company. *International Journal of Computer Applications in Technology*, vol. 45, no 4, pp. 220–230.
16. Beklaryan A.L., Akopov A.S. (2018) Simulation model of the optimal allocation of credit applications for interregional underwriting center of a commercial bank. *Vestnik komp'uternykh i informatsionnykh tekhnologii*, no 11, pp. 45–56 (in Russian).
17. Akopov A.S., Beklaryan L.A. (2015) An agent model of crowd behavior in emergencies. *Automation and Remote Control*, vol. 76, no 10, pp. 1817–1827.
18. Beklaryan A.L., Akopov A.S. (2016) Simulation of agent-rescuer behaviour in emergency based on modified fuzzy clustering. Proceedings of the *2016 International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016)*. Singapore, 9–13 May 2016, pp. 1275–1276.
19. Zhang Q., Liu W., Li H. (2009) The performance of a new version of MOEA/D on CEC09 unconstrained MOP test instances. Proceedings of the *2009 IEEE Congress on Evolutionary Computation (CEC 2009)*. Trondheim, Norway, 18–21 May 2009, pp. 203–208.
20. Deb K., Pratap A., Agarwal S., Meyarivan T. (2000) A fast elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, vol. 6, no 2, pp. 182–197.
21. Jain H., Deb K. (2014) An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach. Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, vol. 18, no 4, pp. 602–622.
22. Zhong W, Liu J, Xue M., Jiao L. (2004) A multiagent genetic algorithm for global numerical optimization. *IEEE Transactions on Systems, Man, and Cybernetics. Part B (Cybernetics)*, vol. 34, no 2, pp. 1128–1141.
23. Deb K., Kumar A. (1995) Realcoded genetic algorithms with simulated binary crossover: Studies on multimodal and multiobjective problems. *Complex Systems*, vol. 9, no 6, pp. 431–454.
24. Deep K., Thakur M. (2007) A new crossover operator for real coded genetic algorithms. *Applied Mathematics and Computation*, vol. 188, no 1, pp. 895–911.
25. Deep K., Thakur M. (2007) A new mutation operator for real coded genetic algorithms. *Applied Mathematics and Computation*, vol. 193, no 1, pp. 211–230.

About the authors

Andranik S. Akopov

Dr. Sci. (Tech.);

Professor, Department of Business Analytics, National Research University Higher School of Economics, 20, Myasnitskaya Street, Moscow 101000, Russia;

Chief Researcher, Laboratory of Dynamic Models of Economy and Optimization, Central Economics and Mathematics Institute, Russian Academy of Sciences, 47, Nakhimovky Prospect, Moscow 117418, Russia;

E-mail: aakopov@hse.ru

Armen L. Beklaryan

Cand. Sci. (Tech.);

Associate Professor, Department of Business Analytics, National Research University Higher School of Economics, 20, Myasnitskaya Street, Moscow 101000, Russia;

Senior Researcher, Laboratory of Social Modeling, Central Economics and Mathematics Institute, Russian Academy of Sciences, 47, Nakhimovky Prospect, Moscow 117418, Russia;

E-mail: abeklaryan@hse.ru

Manoj Thakur

PhD;

Associate Professor, School of Basic Sciences, Indian Institute of Technology Mandi, Mandi, Himachal Pradesh 175005, India;

E-mail: manoj@iitmandi.ac.in

Bhisham Dev Verma

Doctoral Student, School of Basic Sciences, Indian Institute of Technology Mandi, Mandi, Himachal Pradesh 175005, India;

E-mail: bhishamdevverma@gmail.com