DOI: 10.17323/2587-814X.2025.1.34.49

Modeling and optimization of the characteristics of intelligent transport systems for "smart cities" using hybrid evolutionary algorithms

Evgeniy A. Zaripov
E-mail: e.a.zaripov@ya.ru

Andranik S. Akopov
E-mail: akopovas@umail.ru

Central Economic Mathematical Institute, Moscow, Russia
MIREA – Russian Technological University, Moscow, Russia

Abstract

Modern cities are facing increasing traffic congestion, necessitating the implementation of intelligent traffic management systems. One of the key areas in this field is adaptive traffic signal control, which can adjust to changing traffic conditions. However, existing methods for optimizing traffic signal cycle parameters have several limitations, such as high computational complexity, the risk of premature convergence of algorithms and the difficulty of accounting for traffic dynamics. This study proposes an approach to optimizing the characteristics of an intelligent transportation system using hybrid evolutionary algorithms. The methods we developed combine the principles of genetic algorithms (GA) and particle swarm optimization (PSO), enabling a balance between global and local search for optimal parameters. The research examines six different hybridization schemes, including modified versions of basic algorithms, as well as their integration with HDBSCAN clustering methods for adaptive optimization frequency tuning. To evaluate the effectiveness of the proposed algorithms, a simulation model was developed in the AnyLogic environment, replicating real urban traffic conditions. Numerical experiments conducted on a local section of the road network in Moscow demonstrated that the hybrid SlipToBest algorithm achieves the best results in reducing average travel time and fuel consumption, while the Alternating algorithm (structured switching between GA and PSO) ensures high solution stability. The results of this study confirm the feasibility of using hybrid evolutionary methods for traffic flow management tasks. The proposed algorithms not only enhance the efficiency of traffic signal control but also establish a foundation for the further development of adaptive urban traffic management systems.

Keywords: intelligent transport systems, traffic infrastructure management, smart city, hybrid evolutionary algorithms, simulation modeling, traffic management, AnyLogic

Citation: Zaripov E.A., Akopov A.S. (2025) Modeling and optimization of the characteristics of intelligent transport systems for "smart cities" using hybrid evolutionary algorithms. *Business Informatics*, vol. 19, no. 1, pp. 34–49. DOI: 10.17323/2587-814X.2025.1.34.49

Introduction

In the context of growing urbanization, adaptive traffic flow management has become an essential tool for improving road network efficiency. One of the key areas in this regard is traffic signal optimization, which enables dynamic phase adjustments based on traffic intensity. However, determining the optimal parameters for traffic signal cycles presents a complex optimization problem characterized by high dimensionality, nonlinearity and stochastic variations in traffic flows.

Traditional analytical methods and heuristic algorithms often prove ineffective in solving such problems due to their inability to adapt to changing traffic conditions and their high computational complexity. As a result, increasing attention has been given in recent years to evolutionary algorithms such as the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). These methods allow for approximate optimal solutions without requiring gradient calculations, making them particularly attractive for optimization problems in complex systems.

Despite their proven effectiveness, both approaches have limitations. GA provides good solution diversification but often converges slowly. PSO, on the other hand, demonstrates rapid convergence but is prone to premature convergence to local minima. To overcome these shortcomings, hybrid methods that combine the strengths of GA and PSO are actively being developed. Hybridization allows for a balance between global and local search processes, which is particularly important for dynamic systems such as traffic signal control.

This study proposes an improved hybrid evolutionary optimization method for traffic signal regulation. Various integration schemes of GA and PSO are considered, aimed at enhancing the accuracy and stability of solutions. A key element of the proposed approach is a modified swarm algorithm, "SlipToBest," which introduces a directed shift of particles toward the best-found solution to accelerate the optimization process. Additionally, the "Alternating" method is analyzed, which involves alternating GA and PSO to achieve a balance between global and local search strategies.

Furthermore, the study explores the potential application of the HDBSCAN clustering algorithm for analyzing traffic flow density and dynamically adjusting optimization frequency. This reduces computational costs and enhances the adaptability of the algorithms to changing traffic conditions.

The effectiveness of the methods we developed is evaluated using a traffic flow simulation model built in the AnyLogic environment. The experiments are conducted using real-world data from the road network of Moscow, allowing for an objective assessment of the proposed algorithms' impact on average travel time and fuel consumption.

The results demonstrate that the proposed hybrid methods significantly outperform classical algorithms. SlipToBest shows the best performance in terms of convergence speed and travel time reduction, while Alternating ensures solution stability even under high traffic variability. The inclusion of the HDBSCAN clustering method further enhances the adaptability of the control system.

Thus, the study confirms the effectiveness of hybrid evolutionary algorithms for intelligent traffic flow management and highlights the necessity for their further improvement to enhance the quality of urban transportation infrastructure.

1. Formulation of the optimization problem for an intelligent transportation system

The optimization of traffic signal control parameters is complicated by high dimensionality, nonlinearity and the stochastic nature of traffic flows, as confirmed by research findings [1–5]. Under such conditions, classical methods, including gradient-based algorithms and heuristics, often require significant model simplifications, reducing the accuracy of the solutions obrained. At the same time, evolutionary algorithms demonstrate the ability to perform global search and adapt to changing conditions [4, 7, 9], making them preferable for solving such problems. The most widely used evolutionary optimization methods are Genetic Algorithms (GA), which utilize selection, crossover, and mutation mechanisms to search for optimal solutions, and Particle Swarm Optimization (PSO), which simulates the collective behavior of agents in the search space [3–6]. GA effectively explores the solution space and provides high diversity but may exhibit slow convergence. In contrast, PSO achieves high search speed but is prone to premature convergence to local minima [7, 10–12].

To overcome these limitations, this study proposes a hybridization of GA and PSO that combines their strengths. Several hybridization schemes are considered, including **Alternating** (alternating GA and PSO stages), **SlipToBest** (an additional shift towards the best-found solution in PSO), **MixIntegrate** (integration of genetic operators into the swarm search process), and **Mix** (a simple combination without a rigid structure). As demonstrated in several studies, hybrid methods improve search efficiency, enhancing both convergence speed and solution quality [7, 10–12].

The optimization of traffic signal control parameters in a multi-agent transportation system is formulated as a minimization problem of an objective function that defines the quality of the transportation process. Let $X = (x_1, x_2, \ldots, x_n)$ be the parameter vector, including the durations of red and green traffic light phases. Constraints on phase parameters are defined in accordance with technical regulations, determining the search space: $x_i \in [L_i, U_i]$, $i = 1, 2, \ldots, n$, where L_i, U_i are the lower and upper allowable durations of traffic signals, and n is the total number of controlled signals.

The objective function is defined as the average travel time of vehicles passing through the studied road section during a given simulation period. Let this value be denoted as $f(x)$. Its computation is performed using a simulation model in the AnyLogic environment, which replicates vehicle movement for each parameter set and records travel time and fuel consumption indicators. The average travel time serves as an integral characteristic of traffic flow quality, as it reflects the impact of traffic intensity, intersection throughput and traffic signal control logic [3–5].

In addition to average travel time, additional indicators such as fuel consumption and other economic and environmental factors are considered. However, the primary optimization goal is to minimize travel time, as it is the key criterion for evaluating the efficiency of traffic signal control. This formulation allows for an objective assessment of the impact of different algorithms on traffic processes and ensures a correct comparison of their effectiveness. Thus, the problem is formally defined as:

$$\min_{x_i \in X} f(x_i). \tag{1}$$

subject to the constraints:

$$x_i \in [L_i, U_i],\ i = 1, 2, \ldots, n.$$

This problem formulation is widely used in transportation system studies [1, 2] since minimizing average travel time directly correlates with improving network throughput, reducing congestion and optimizing fuel consumption. The use of this objective function is justified by fundamental studies on evolutionary adaptation [4] and research in transportation optimization [7, 9]. It has been proven that evolutionary algorithms, including GA and PSO, efficiently minimize such functions without requiring analytical gradients, allowing for the modeling of complex dynamic processes [3–6, 9]. Thus, the minimization of this function serves as the primary criterion for selecting the optimal set of traffic signal control parameters, ensuring shorter travel times and lower transportation costs.

The problem (1) can, in particular, be solved using the **PSO algorithm**, due to its ability to efficiently search for solutions without requiring gradient information [3, 13]. This approach simulates the collective behavior of particles that exchange information about discovered improvements and move toward the global optimum.

Each particle in PSO has a current position, a velocity that defines the rate of parameter change, a personal best position, and a best global position. At each iteration, particles adjust their velocity and position based on acquired information using update procedures [5, 6, 13, 14]. One of the algorithms used in this study (**Swarm**) is also implemented based on this classical PSO framework. This algorithm serves as a baseline for comparing the effectiveness of hybrid modifications. Shi and Eberhart [6] proposed a strategy of dynamically changing the inertia weight (variable w) to balance between exploration of the solution space and exploitation of already found solutions. Clerc and Kennedy [5] introduced the concept of constriction coefficients, ensuring stable convergence. Recent studies continue to explore optimal parameter settings for various problem classes, including transportation optimization [1, 2, 9, 14, 15]. PSO has proven to be an effective global search method, demonstrating successful applications in transportation optimization tasks.

Additionally, problem (1) can be solved using the **Genetic Algorithm**, an evolutionary optimization method based on selection, crossover and mutation mechanisms that mimic the natural evolution process. Initially proposed by Holland [4], GA has demonstrated efficiency in optimizing complex, high-dimensional search spaces [7, 16–18]. In this study, GA is used to optimize traffic signal control parameters, where each individual represents a parameter vector $x = (x_1, x_2, \ldots, x_n)$, defining the durations of traffic light phases.

The initial population is generated randomly or using heuristics, followed by iterative selection, crossover and mutation processes to improve solutions. The crossover operation involves combining parameters from two parent individuals to create offspring. Mutation represents a random modification of individual parameters, preventing premature convergence of the algorithm. Mutation plays a crucial role in exploring new regions of the search space and enhancing solution quality. GA ensures a global search capability, allowing for effective exploration of complex search spaces. In this study, a classical GA implementation (**Parallel**) is considered, which employs basic crossover and mutation mechanisms. This algorithm serves as a foundation for further hybrid modifications that combine GA with PSO.

The third and most promising approach to solving problem (1) is based on the **hybridization of PSO and GA**. This hybridization leverages the strengths of both methods: GA's ability to perform global search and overcome local minima, along with PSO's capability to rapidly refine solutions and move toward optimal points in the search space. Such a combination improves both convergence speed and solution quality, especially in complex dynamic systems such as transportation infrastructure [1, 2, 9, 10].

This study considers four implemented hybrid schemes, differing in the principle of operator combination and optimization strategy.

- **Alternating (GA and PSO alternation).** This approach involves the sequential application of GA and PSO at different iterations of the algorithm. Let the number of iterations be defined as ITERATIONS. On even steps, GA is applied, while on odd steps, PSO is used (or vice versa, depending on the initial conditions):

$$\begin{aligned} &iter \bmod 2 = 0 \Rightarrow apply\ GA, \\ &iter \bmod 2 = 1 \Rightarrow apply\ PSO, \end{aligned} \tag{2}$$

Such combination strategies have been mentioned in reviews on hybrid methods [9, 19–21]. Alternating helps maintain a balance between global and local search: GA periodically updates solutions, increasing diversity, while PSO efficiently refines candidate solutions.

- **SlipToBest (PSO + alpha).** This is a proposed PSO modification in which, after updating their positions, each particle is additionally shifted toward the globally best solution by a parameter α. That is, the value of the j-th target variable of the i-th particle can be computed as:

$$x_{i,j}(t+1) = x_{i,j}(t) + \alpha\left(g_j^{best}(t) - x_{i,j}(t)\right), \tag{3}$$

which defines a directed shift toward the best solution. This method accelerates convergence but may reduce solution diversity if the parameter α is set too high [6, 11]. Here,

$x_{i,j}$ is the value of the j-th target variable of the i-th particle at iteration t;

$g_j^{best}(t)$ is the globally best value of the j-th target variable found by all swarm particles up to iteration t.

- **MixIntegrate (Genetic Operators within PSO).** This approach **integrates genetic operators** (crossover and mutation) directly into the swarm update stage. After executing the standard PSO step, genetic operators are applied to **selected particle pairs**:

$$x_{j,new} = crossover(x_{i,j}, \bar{x}_{\bar{i},j}),\ x'_{j,new} = mutate(x_{i,new}), \tag{4}$$

where

$\{x_{j,new}, x'_{j,new}\}$ are the new values of the j-th target variable obtained from the crossover and mutation operations;

$\{x_{i,j}, \bar{x}_{\bar{i},j}\}$ are the values of the j-th target variable for the i-th and $\bar{i}$-th particles in the swarm.

  - **The integration of genetic operators into PSO** helps maintain solution diversity and prevents premature convergence of the swarm. Similar strategies have been discussed in hybrid optimization research [7, 9].

- **Mix (Simple Combination of PSO and GA).** This is a less structured hybridization approach in which PSO and GA are applied without fixed rules or systematic use of operators: Mix: Apply PSO, then GA, or vice versa, without a fixed rule.
  - **A simple Mix scheme**. This approach has been mentioned in the literature as an example of unstructured hybridization, which does not always outperform pure PSO or GA [10, 12, 20].

To assess traffic flow density, this study employs the **HDBSCAN** (Hierarchical Density-Based Spatial Clustering of Applications with Noise) hybrid clustering algorithm [21], which extends the DBSCAN method hierarchically. This algorithm identifies groups of points (clusters) based on their density and automatically determines the optimal number of clusters, making it more flexible compared to classical methods such as k-means.

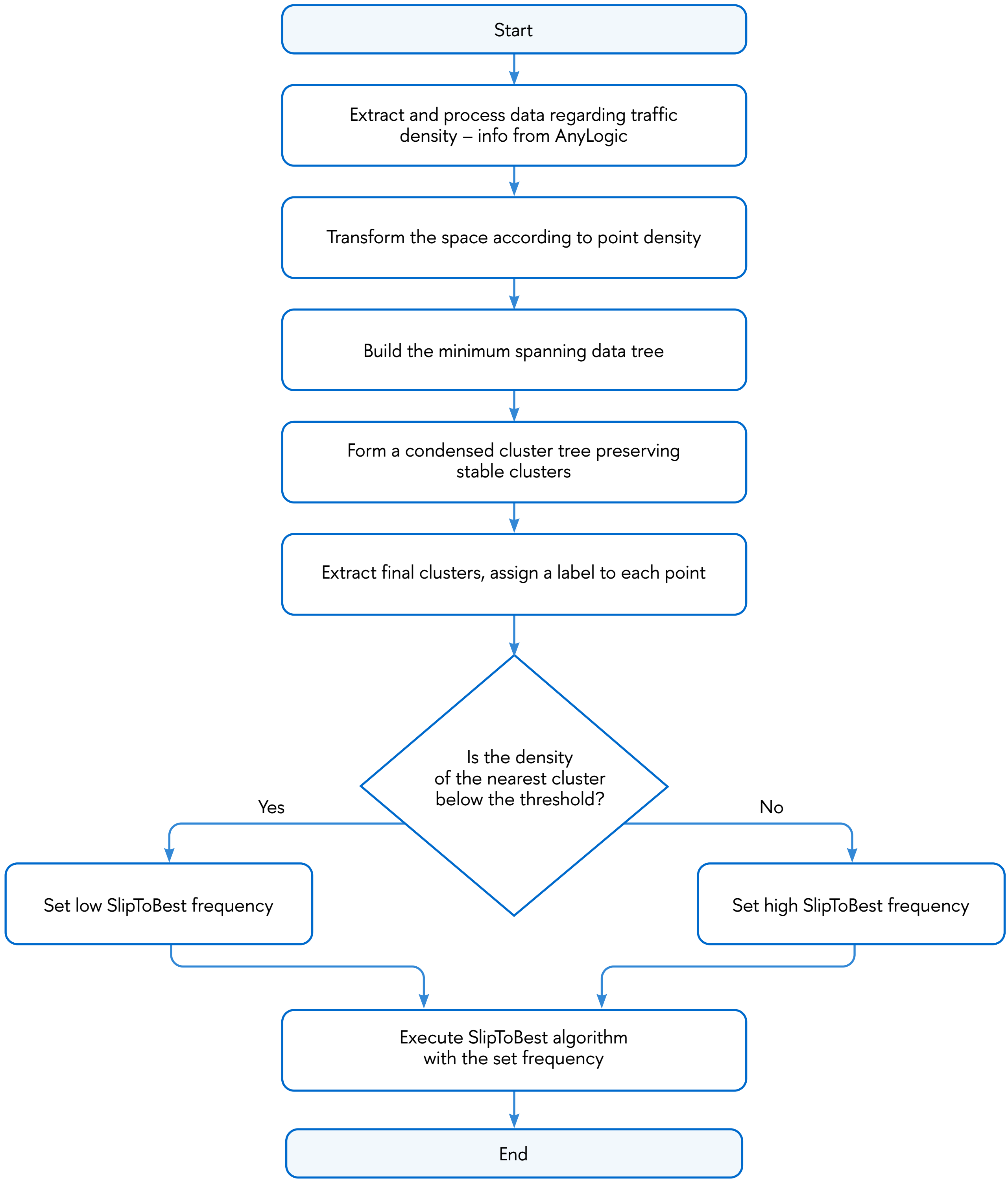


Fig. 1. High-level workflow diagram of the HDBSCAN algorithm integrated with the AnyLogic simulation model.

The key advantage of HDBSCAN is its ability to account for the dynamic structure of traffic flows, automatically detecting areas of high and low traffic density. This is particularly important for traffic analysis, where movement intensity can fluctuate significantly depending on the time of day, road infrastructure and unforeseen factors (e.g., accidents, roadworks). The algorithm also identifies noise points, which may correspond to anomalous traffic situations such as congestion or sudden traffic pattern changes.

To integrate HDBSCAN with the AnyLogic simulation model, a workflow diagram of the algorithm has been developed (*Fig. 1*). At each simulation step (at time t), the following stages are executed:

1. Collecting and processing traffic density data obtained from the "smart" traffic light system.

2. Analyzing the spatial distribution of vehicles around the traffic light, considering their density.

3. Constructing a minimum spanning tree to represent hierarchical cluster structures.

4. Evaluating cluster stability (analyzing density and cluster duration over time).

5. Forming a condensed tree to identify stable clusters.

6. Determining the final cluster structure and assigning appropriate labels.

HDBSCAN enables the adaptive adjustment of the activation frequency of the SlipToBest hybrid algorithm:

- If the nearest cluster to the traffic light has low density, the system reduces the frequency of SlipToBest application (minimizing computational load).
- If the cluster density exceeds a threshold value, SlipToBest is triggered more frequently, ensuring a prompt response to changing traffic conditions.

Thus, the HDBSCAN algorithm dynamically adapts traffic signal phase control, allowing for more efficient resource allocation and a more precise consideration of fluctuating road conditions.

2. Implementation of the simulation model in AnyLogic

For the listed hybrid algorithms, parameters such as the coefficient, the frequency of GA operator application, and the number of PSO iterations between genetic stages are selected empirically or based on recommendations from scientific literature [6, 7, 9–12, 19–22]. To ensure an objective comparison of their effectiveness, a unified simulation environment (AnyLogic) and a common objective function (see (1)) are used, guaranteeing the correctness of the analysis. The use of AnyLogic in adaptive traffic management tasks has previously demonstrated its effectiveness [23].

The AnyLogic model so developed extends the capabilities of classical PSO and GA, providing a flexible and efficient search for optimal traffic signal parameters. The following sections present the results of applying hybrid algorithms and a comparative analysis of their effectiveness.

During the refinement of the simulation model, improvements were made to bring it closer to real-world traffic conditions and to simplify integration with external optimization algorithms. Vehicles now switch lanes considering lane priority, the speed of adjacent vehicles, and the distance to stop lines. These modifications have increased the realism of the simulation, making the behavior of transport agents closely resemble real urban environments.

The model's variables and parameters, including the durations of red and green traffic light phases, were structured in a way that allows them to be easily read, modified and analyzed from external code. This significantly simplified integration with evolutionary algorithms and enabled mass iterations to assess the quality of traffic management decisions.

The model's structure and software implementation were optimized so that adding or removing traffic signals does not require significant changes to the source code. This flexibility allows the model to be adapted to different scenarios, modify the road network topology and adjust the optimization area without substantial time costs.

The developed model is prepared for integration with external optimization code. Its modular architecture allows AnyLogic to be used in a "black box" mode, where algorithms receive traffic signal parameters, process them, and return optimized values that minimize average travel time and reduce fuel consumption. This approach makes the implementation of new optimization algorithms convenient and ensures an objective comparison of their performance.

Figure 2 presents the structure of the updated model of an urban road network segment created in AnyLogic. This model includes the placement of traffic signals and phase timing parameters, allowing for flexible testing of various adaptive control algorithms.

The simulation model allows for variations in key traffic signal control parameters that directly affect network capacity and traffic dynamics. *Figure 3* illustrates the parameter panel of a typical "smart" traffic light (trafficLight4).

The model defines phase durations (*redTime*, *greenTime*), links to stop lines and the traffic light's operating mode. Depending on traffic intensity and intersection configuration, the model adjusts signal switching logic, determining optimal green and red phase intervals.

During experiments, the algorithm generates control decisions, determining the optimal $redTime_i$ and $greenTime_i$ for all traffic signals. These parameters are loaded into the model before the simulation starts. Then, AnyLogic simulates vehicle flow, recording average travel time and fuel consumption. Thus, the evolutionary algorithm receives an objective evaluation of control quality, analyzing the impact of parameters on the objective function.

In subsequent iterations, the algorithm updates traffic signal parameters, minimizing the average travel time.

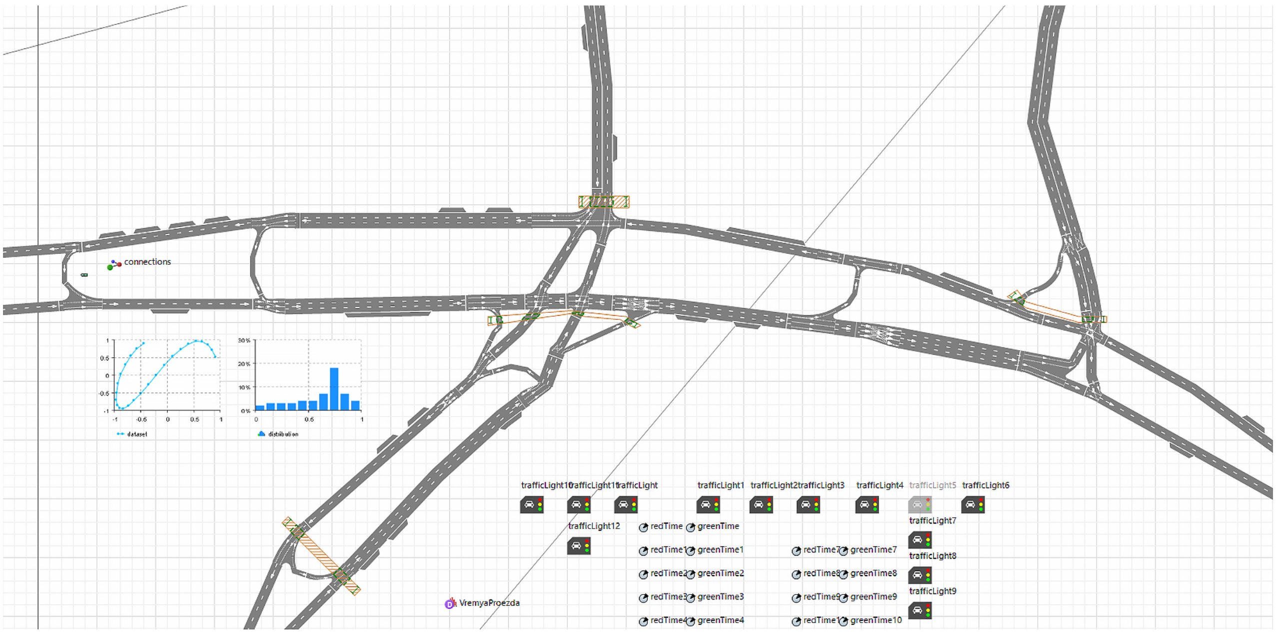


Fig. 2. Structure of the updated simulated urban road network segment in AnyLogic with placed traffic signals and phase timing parameters.

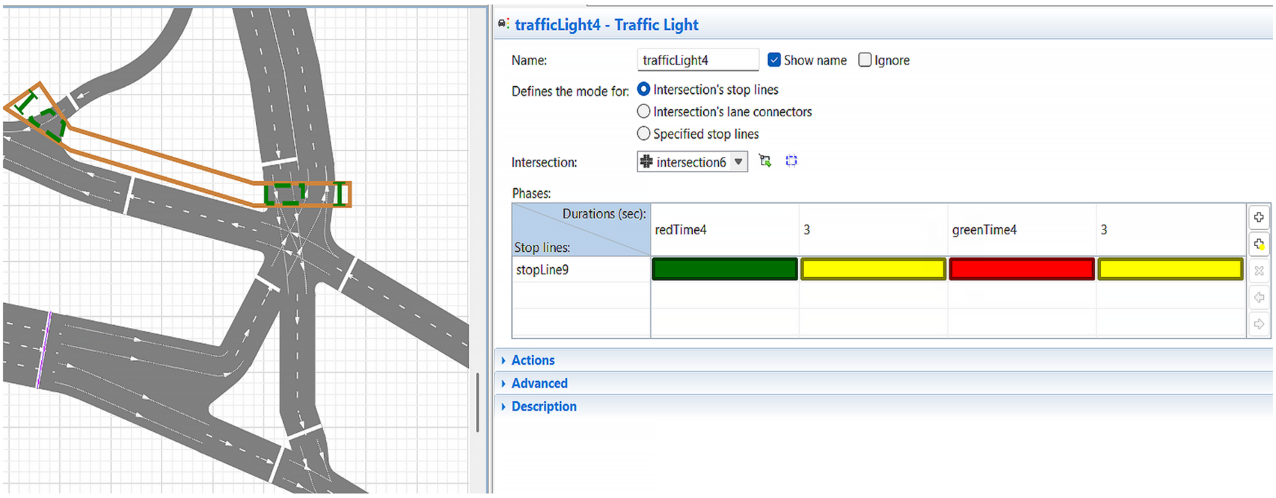


Fig. 3. Parameter panel of the "smart" traffic light (trafficLight4) in AnyLogic with phase durations and an attached stop line.

The model automatically adapts to parameter changes, allowing for dynamic traffic signal adjustments to various traffic conditions, including light traffic and heavy congestion.

3. Results of numerical experiments and analysis of algorithm efficiency

The traffic flow simulation model developed in this study for a local urban road network segment in AnyLogic was previously described in detail in the authors' works [1, 2]. The model considers the characteristics of road infrastructure, the placement of traffic lights and pedestrian crossings, and is based on an agent-based and discrete-event approach. The simulation parameters, including simulation duration, road network configuration and traffic flow intensity, were determined based on real traffic data near the Yugo-Zapadnaya metro station in Moscow [1].

During the experiments, the variable parameters were the durations of the red and green traffic light phases, as these characteristics have a critical impact on the network's capacity. Each phase configuration was evaluated based on the average travel time of vehicles and fuel consumption indicators, which were computed using simulation results.

For each of the examined algorithms (Alternating, SlipToBest, Swarm, Parallel, MixIntegrate, Mix), multiple simulation runs were conducted to reduce the influence of stochastic factors and assess solution stability. The stopping criterion was most often a fixed number of iterations or generations determined by computational resource constraints and empirical observations. Additionally, a threshold for lack of improvement was considered, beyond which the algorithm was terminated.

The analyzed algorithms differ in their structural composition and the way they combine swarm-based (PSO) and genetic (GA) approaches, which affects their effectiveness. To ensure an objective comparison, the following metrics were used:

- Average travel time, which characterizes the road segment's capacity and the level of convenience for drivers.
- Fuel consumption (AllLostGas), reflecting the economic and environmental efficiency of the system, as reducing unnecessary vehicle idling leads to lower fuel consumption.

Table 1.

Comparative results for average travel time and fuel consumption

Algorithm	Approach characteristics	Best average travel time (min)	Fuel consumption (L)
SlipToBest	Swarm algorithm with an additional shift toward the best solution (alpha parameter)	3.2213	288.5
Alternating	Alternating GA and PSO	3.2483	295.4
Swarm	Pure particle swarm optimization (PSO)	3.2246	296.2
Mix	Simple combination of PSO and GA, without strict alternation	3.2650	308.7
MixIntegrate	Swarm algorithm with GA operators (mutation, crossover)	3.2798	312.9
Parallel	Pure genetic algorithm (GA)	3.3009	318.5

The results of numerical experiments are summarized in *Table 1*, which presents the best average travel times achieved by each algorithm. *Figure 4* provides a graphical visualization of the effectiveness of the examined approaches.

Fuel consumption refers to the total excess fuel consumption (liters) for a given set of vehicles (e.g., 10 000 vehicles) participating in multiple simulation runs. The numerical values were averaged to account for stochastic fluctuations in road traffic, which arise from variations in acceleration, braking, and traffic density.

The analysis of data from *Table 1* shows that SlipToBest demonstrates the best results, achieving the lowest average travel time with minimal fuel consump-

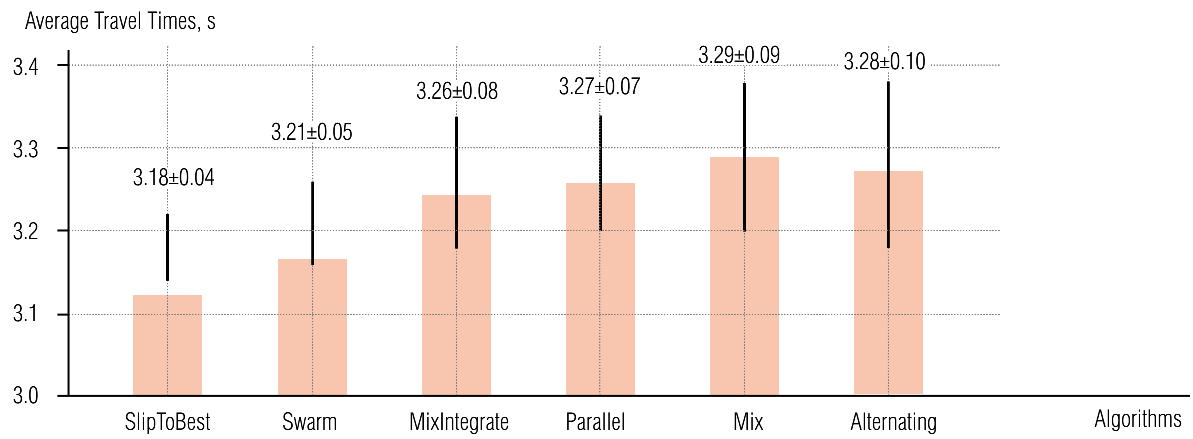


Fig. 4. Comparison of algorithm efficiency based on average travel time.

tion. The next most effective algorithms are Alternating and Swarm. Notably, Alternating, due to its alternating application of GA and PSO operators, outperforms both pure PSO (Swarm) and less structured hybrid approaches (Mix, MixIntegrate). The use of a pure genetic algorithm (Parallel) leads to the highest time and fuel consumption, indicating its insufficient efficiency without additional modifications.

Thus, targeted improvements to the particle swarm algorithm (such as introducing the alpha parameter in SlipToBest) or rational alternation of evolutionary operators (as in Alternating) provide the greatest gains in travel speed and fuel cost reduction. At the same time, unstructured hybridization schemes (Mix, MixIntegrate) and the pure genetic algorithm (Parallel) lag behind more organized hybrid architectures, confirming the importance of a systematic approach when designing evolutionary algorithms.

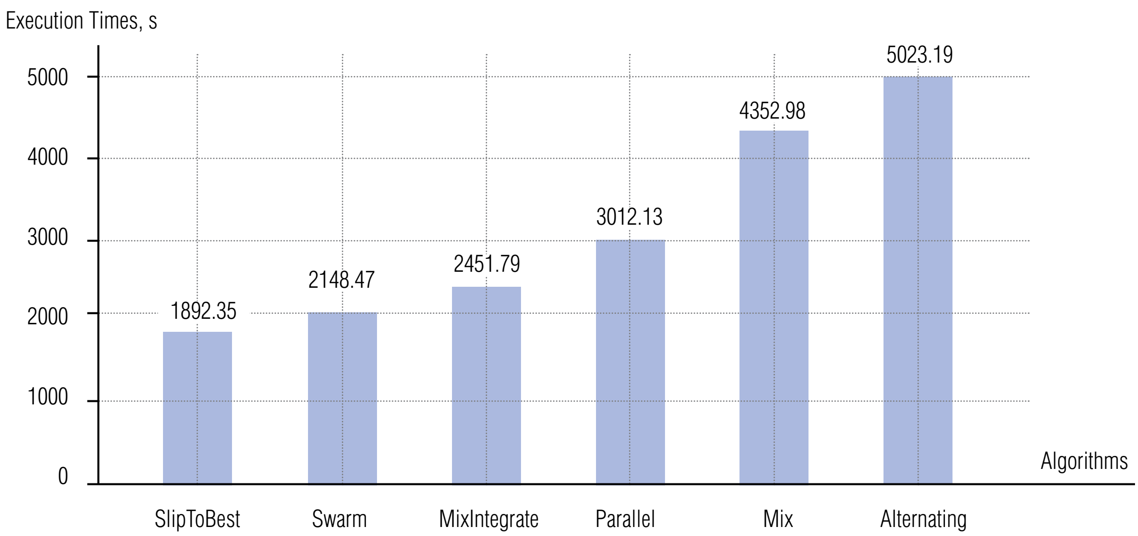
To provide a more detailed comparison, an assessment of the execution time of different algorithms for traffic signal cycle control on the studied road segment was conducted. The results are presented in numerical format, allowing for a comparison of the computational complexity of the methods and their applicability in practical modeling scenarios.

The experiment analyzed six optimization algorithms:

- **SlipToBest** – a modified particle swarm optimization algorithm with the *alpha* parameter.
- **Parallel** – a pure genetic algorithm with parallel data processing.
- **Swarm** – a standard implementation of PSO.
- **MixIntegrate** – a combined method integrating PSO with genetic operators (mutation, crossover).
- **Mix** – a hybrid algorithm combining PSO and GA without a strict operator alternation scheme.
- **Alternating** – a method in which PSO and GA stages are alternated to balance global and local search.

The primary goal of the analysis was to determine the execution time of each algorithm under fixed input parameters in the AnyLogic environment.

Figure 5 presents the experiment results, illustrating the execution time of the algorithms in seconds based on numerical simulations.



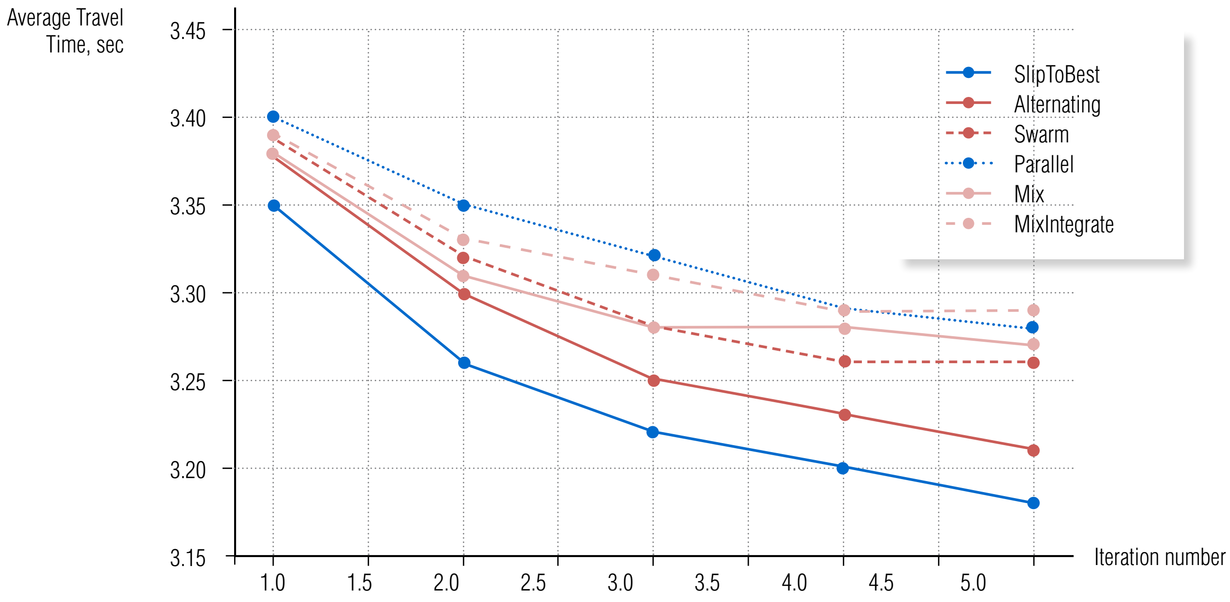


Fig. 6. Convergence dynamics of the average travel time.

The analysis of the data obtained shows that SlipToBest has the shortest execution time – 1992.78 seconds, indicating the high computational efficiency of the algorithm. Optimization of the α parameter allowed for accelerated convergence and reduced overall computation time. Parallel demonstrated 2415.93 seconds due to parallel data processing, significantly speeding up calculations compared to sequential methods. Swarm (classical PSO) recorded 3008.97 seconds, confirming its stable but not the fastest convergence compared to the modified SlipToBest.

The MixIntegrate and Mix algorithms required 4591.87 and 6351.96 seconds, respectively, which can be attributed to their complex hybrid structure and the additional iterations required for mutation and crossover operations. Alternating, which alternates between PSO and GA, exhibited the longest execution time – 6542.57 seconds, due to its high computational complexity, despite the high accuracy of the obtained solutions.

Analyzing the results, SlipToBest stands out as the most efficient in terms of computational speed, making it the optimal choice for real-time traffic flow management. Parallel and Swarm provide balanced results and can be used for practical optimization problems, where a trade-off between computational complexity and solution quality is needed. MixIntegrate and Mix, while delivering more accurate solutions, require computational load optimization to enhance their practical applicability. Alternating, despite producing the most precise solutions, demands significant computational resources, making it more suitable for tasks where accuracy is prioritized over execution speed.

These findings highlight the importance of choosing an appropriate algorithm depending on the modeling scenario. Future research should focus on enhancing hybrid methods to achieve an optimal balance between computational efficiency and solution quality.

During the experiments, each algorithm was tested under different initial conditions, including randomly initialized swarms for PSO and varied initial populations for GA. The results indicate that SlipToBest not only achieves the best average travel time but also exhibits high solution stability, showing minimal variability under different initial parameter settings.

Figure 6 illustrates the convergence dynamics of the average travel time $\overline{T}$. As shown in the graph, SlipToBest reaches the minimum value of $\overline{T}$ faster than other algorithms, thanks to the directed shift of swarm particles toward the globally best solution. Alternating also demonstrates high solution quality; however, in most repeated experiments, SlipToBest shows a faster approach to the optimum.

The remaining algorithms (pure PSO – Swarm, pure GA – Parallel, as well as unstructured hybrids Mix and MixIntegrate) could not outperform SlipToBest in terms of convergence speed or solution stability.

To formally validate these differences, a statistical analysis was conducted, including:

- Comparison of mean values $\overline{T}$.
- Analysis of variance (ANOVA).
- Nonparametric tests with a significance level of $p < 0.05$.

The results of the statistical analysis confirm the superiority of SlipToBest in both average travel time and solution stability.

When developing hybrid evolutionary algorithms, the tuning of key parameters plays a crucial role, including:

- PSO coefficients (w, c_1, c_2) and swarm size.
- The parameter α in SlipToBest, which determines the intensity of the swarm's attraction to the best-found solution.
- GA parameters (*mutationRate, crossoverRate*) and population size in hybrid schemes.

Experimental data show that pure PSO and GA often underperform in certain cases, and improper operator combination (as seen in Mix and MixIntegrate) can lead to inefficient allocation of computational resources. The SlipToBest modification addresses this issue by enhancing the swarm's ability for fast and stable search.

The experimental results lead to the following key conclusions:

- SlipToBest is the optimal choice for optimization in complex dynamic systems, where fast convergence to a high-quality solution is critical.
- Tuning the α parameter in SlipToBest (typically within the range 0.1–0.3) allows for a flexible trade-off between search speed and accuracy.
- With limited computational resources, it is essential to consider that SlipToBest demonstrates one of the best execution speeds (see *Fig. 4*), ensuring minimum travel time compared to Alternating.
- A targeted modification of the swarm algorithm leads to superior performance in the average metric $\overline{T}$ and enhances solution stability.
- Simple hybridization (Mix, MixIntegrate) does not guarantee superiority over pure algorithms, whereas a well-structured alternation strategy (Alternating) or a directed step toward the best solution (SlipToBest) significantly improves method efficiency.

Thus, properly structured hybrid algorithms can outperform pure PSO and GA, providing an optimal balance between convergence speed and solution quality. However, an irrational combination of PSO and GA operators can reduce effectiveness, as observed in the Mix and MixIntegrate cases.

Conclusion

In this study, a new optimization method for intelligent transportation systems (ITS) has been developed and tested, utilizing hybrid evolutionary algorithms aimed at improving traffic signal control efficiency in local urban road network segments. To conduct the experiments, an agent-based model was created in AnyLogic, simulating real traffic conditions. The Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) were used as the baseline methods, forming the foundation for the proposed hybrid schemes.

Based on these methods, six algorithmic frameworks were designed and analyzed: Alternating, SlipToBest, MixIntegrate, Mix, as well as pure PSO (Swarm) and pure GA (Parallel). The primary evaluation metric was average travel time, supplemented by fuel consumption analysis, allowing for the consideration of both economic and environmental factors.

The results of comparative experiments lead to the following conclusions:

1. SlipToBest, a modified PSO with an additional "pulling" mechanism that directs particles toward the best solution, demonstrated the highest performance in both convergence speed and solution quality.

2. Alternating, which systematically alternates GA and PSO, performed slightly worse than SlipToBest but provided more stable solutions due to its balanced use of global and local search.

3. Swarm (baseline PSO) outperformed Parallel (pure GA) but fell behind the leading methods (SlipToBest and Alternating).

4. Unstructured hybrids (MixIntegrate and Mix) failed to outperform the implemented methods, highlighting the importance of a well-designed hybridization strategy.

Thus, a simple combination of GA and PSO without a structured strategy does not provide significant advantages. High performance is achieved through specific modifications, such as the directed particle shift in PSO (SlipToBest) or structured alternation of GA and PSO operators (Alternating).

The theoretical significance of this research lies in the advancement and refinement of hybrid evolutionary algorithms for intelligent transportation systems. The practical value is in the potential real-world application of the proposed solutions for traffic signal control, which can reduce average travel time, lower fuel consumption and reduce harmful emissions.

Research limitations include the local scale of the simulated road network and certain simplifying assumptions regarding driver and pedestrian behavior. Future research will focus on expanding the modeling scope, considering seasonal and weather factors, and integrating adaptive parameter tuning strategies for evolutionary algorithms with deep learning and reinforcement learning techniques.

A promising research direction is the development of hybrid algorithms capable of adapting to dynamically changing traffic conditions. This will further enhance transportation infrastructure and provide long-term socio-economic benefits for urban traffic systems. ■

Acknowledgments

This research was supported by the Russian Science Foundation grant (Project No. 23-11-00080).

References

1. Zaripov E.A., Melnikov A.M., Akopov A.S. (2024) Simulation modeling and optimization of traffic flows in local sections of the street-road network using AnyLogic. *Information Technologies*, vol. 30, no. 4, pp. 183–189. https://doi.org/10.17587/it.30.183-189.

2. Akopov A.S., Zaripov E.A., Melnikov A.M. (2024) Adaptive control of transportation infrastructure in an urban environment using a real-coded genetic algorithm. *Business Informatics*, vol. 18, no. 2, pp. 48–66. https://doi.org/10.17323/2587-814X.2024.2.48.66

3. Eberhart R., Kennedy J. (1995) Particle swarm optimization. Proceedings of the *IEEE International Conference on Neural Networks, Perth, Australia*, pp. 1942–1948.

4. Holland J.H. (1992) *Adaptation in natural and artificial systems*, MIT Press.

5. Clerc M., Kennedy J. (2002) The particle swarm – explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73.

6. Shi Y., Eberhart R. (1998) A modified particle swarm optimizer. Proceedings of the *IEEE International Conference on Evolutionary Computation*, IEEE, pp. 69–73.

7. Bäck T., Fogel D.B., Michalewicz Z. (2000) *Evolutionary computation 1: Basic algorithms and operators*, CRC Press.

8. Wahde M. (2008) *Biologically inspired optimization methods: An introduction*, WIT Press.

9. Coello C.A.C., Lamont G.B., Van Veldhuizen D.A. (2007) *Evolutionary algorithms for solving multi-objective problems*, Springer.

10. Yoshida H., Kawata K., Fukuyama Y., Takayama S., Nakanishi Y. (2000) A particle swarm optimization for reactive power and voltage control considering voltage security assessment // *IEEE Transactions on Power Systems*, vol. 15, no. 4, pp. 1232–1239. https://doi.org/10.1109/59.898095

11. Zhang Y., Wang S., Ji G. (2015) A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, pp. 1–38.

12. Ali M.Z., Pant M. (2009) Improving the performance of particle swarm optimization using differential evolution. Proceedings of the *IEEE Congress on Evolutionary Computation*, pp. 105–112.

13. Poli R., Kennedy J., Blackwell T. (2007) Particle swarm optimization. *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57.

14. Engelbrecht A.P. (2007) *Computational intelligence: An introduction*. 2nd Edition, Wiley.

15. Goldberg D.E. (1989) *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley.

16. Michalewicz Z. (1996) *Genetic algorithms + Data structures = Evolution programs*, Springer.

17. Haupt R.L., Haupt S.E. (2004) *Practical genetic algorithms*, 2nd Edition, Wiley.

18. Tsai P.W., Chiu S.W., Pan J.S., Liao B.Y. (2010) Enhancing particle swarm optimization with multicriterion for engineering optimization. *Expert Systems with Applications*.

19. Ong Y.S., Lim M.H., Zhu N., Wong K.W. (2006) Classification of adaptive memetic algorithms: A comparative study. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*.

20. Zhou A., Qu B.Y., Li H., Zhao S., Suganthan P.N., Zhang Q. (2011) Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and Evolutionary Computation*.

21. Bot D.M., Peeters J., Liesenborgs J., Aerts J. (2023) FLASC: A flare-sensitive clustering algorithm: extending HDBSCAN* for detecting branches in clusters. *arXiv:2311.15887*. https://doi.org/10.48550/arXiv.2311.15887

22. Anfyorov M.A. (2019) Genetic clustering algorithm. *Russian Technological Journal*, vol. 7, no. 6, pp. 134–150 (in Russian). https://doi.org/10.32362/2500-316X-2019-7-6-134-150

23. Beklaryan A.L., Beklaryan L.A., Akopov A.S. (2023) Simulation model of an intelligent transportation system for the 'smart city' with adaptive control of traffic lights based on fuzzy clustering. *Business Informatics*, vol. 17, no. 3, pp. 70–86. https://doi.org/10.17323/2587-814X.2023.3.70.86

About the authors

Evgeniy A. Zaripov

Junior Researcher, Laboratory of Dynamic Economic Models and Optimization, Central Economics and Mathematics Institute, Russian Academy of Sciences, 47, Nakhimovsky Ave., 117418 Moscow, Russia;

PhD Student, Assistant Lecturer, Department of Instrumental and Applied Software, MIREA – Russian Technological University, 78, Vernadsky Ave., 119454 Moscow, Russia;

E-mail: e.a.zaripov@ya.ru

ORCID: 0000-0003-1472-650X

Andranik S. Akopov

Doctor of Technical Sciences, Professor;

Professor of the Russian Academy of Sciences;

Chief Researcher, Laboratory of Dynamic Economic Models and Optimization, Central Economics and Mathematics Institute, Russian Academy of Sciences, 47, Nakhimovsky Ave., 117418 Moscow, Russia;

Professor, Department of Instrumental and Applied Software, MIREA – Russian Technological University, Russia, 78, Vernadsky Ave., 119454 Moscow, Russia;

E-mail: akopovas@umail.ru

ORCID: 0000-0003-0627-3037