ORIGINAL SCIENTIFIC PAPERS

Metaheuristic-based approach to optimizing the weights in the TOPSIS method for driver candidate performance assessment

Ivana M. Sarić[a], *Jasmina* Dž. Vujadinović[b], *Rale* M. Nikolić[c]

[a] University of Belgrade, Faculty of Technology and Metallurgy, Department of Mathematical Sciences, Belgrade, Republic of Serbia; e-mail: isaric@tmf.bg.ac.rs, ORCID iD: https://orcid.org/0000-0001-9141-8098

[b] University of Defence in Belgrade, Military Academy, Department of Natural and Mathematical Sciences, Belgrade, Republic of Serbia; e-mail: jasmina.fijuljanin@va.mod.gov.rs, **corresponding author**, ORCID iD: https://orcid.org/0000-0003-2479-9868

[c] University of Defence in Belgrade, Military Academy, Department of Natural and Mathematical Sciences, Belgrade, Republic of Serbia; e-mail: rale.nikolic@va.mod.gov.rs, ORCID iD: https://orcid.org/0000-0002-8703-3029

doi https://doi.org/10.5937/vojtehg73-62775

FIELD: mathematics, multi-criteria decision making, metaheuristic, cognitive assessment in driver evaluation

ARTICLE TYPE: original scientific paper

Abstract:

Introduction/purpose: Traffic safety and reliable driver selection are the key components of modern transport systems. The aim of this paper is to improve the evaluation process of candidates performance in driving tests by the applying multi-criteria decision-making and metaheuristic approach. Based on the results obtained using the Vienna Test System, a TOPSIS-based model with adaptive weighting of evaluation criteria is proposed.

Methods: The weights of the TOPSIS method were optimized using three metaheuristic algorithms: Genetic Algorithm (GA), Ant Colony Optimization (ACO), and Artificial Bee Colony (ABC) algorithm. Two objective

ACKNOWLEDGMENT: This paper is supported by the Ministry of Defence of the Republic of Serbia under the scientific research project (project name: Driver model in the transport units of the Serbian armed forces , project code: VA-TT/2/20-25).

The authors would like to thank the anonymous reviewers for their valuable comments and constructive suggestions, which significantly improved the quality of this paper.

functions were used during optimization — the AUC and the F1-score — to analyze their impact on model accuracy and stability. The experimental framework consisted of three parts: (1) comparison of GA, ACO, and ABC performance using the AUC as the objective function, (2) analogous comparison using the F1-score as the objective function, and (3) cross-comparison between AUC and F1-score optimized models.

Results: The obtained results indicate that both the choice of metaheuristic algorithm and the objective function significantly influence the performance of the TOPSIS method. AUC-based optimization resulted in more stable models and a better balance between successful and unsuccessful candidates, while F1-based optimization achieved higher sensitivity and better identification of successful candidates.

Conclusions: Applying metaheuristic algorithms for weight optimization within the TOPSIS framework enables adaptive and more reliable candidate ranking, contributing to the development of intelligent driver selection systems and improved traffic safety. The results confirm that an appropriate choice of an optimization algorithm and an objective function can significantly enhance model accuracy and robustness.

Key words: TOPSIS, multi-criteria decision making, metaheuristics, GA, ACO, ABC, Vienna Test System.

Introduction

Traffic safety represents one of the key challenges of modern societies, as road accidents continue to cause significant human, economic, and social consequences worldwide. According to international reports, the human factor remains the dominant cause of most traffic incidents, which emphasizes the importance of systematically assessing drivers abilities and readiness. Consequently, the process of selecting candidates for driving licenses plays a crucial role in improving overall traffic safety levels. In addition to theoretical and practical training, the evaluation of psychomotor, perceptual, and cognitive abilities constitutes an essential part of the driver selection process. In contemporary practice, standardized psychodiagnostic systems based on objective measurement instruments have become highly relevant. Among them, the Vienna Test System (VTS) (Schuhfried, 2013) is one of the most widely used and reliable tools, providing comprehensive assessments of numerous parameters important for safe vehicle operation — including attention, reaction speed, visual perception, risk-taking tendencies, and motor coordination. Such testing results offer valuable input for decision-making regarding candidates driving abilities but simultaneously pose a multi-criteria decision problem, where numerous factors of unequal importance must be objectively combined. In such decision-making contexts, the use of multi-criteria decision-making (MCDM) methods are particularly justified. One of the most prominent among them is the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) (Hwang & Yoon, 1981), recognized for its interpretability and capability to rank alternatives according to their distance from the ideal and anti-ideal solutions (Ren et al., 2021; Huang et al., 2020). However, one of the key challenges in applying TOPSIS lies in determining the weights of criteria, since the final decision outcome depends directly on their distribution. Traditional approaches to assigning weights often rely on subjective expert judgment or predefined assumptions, which may limit the objectivity and accuracy of the model. With advances in computational intelligence, metaheuristic algorithms have emerged as a powerful alternative for optimizing weighting schemes based on empirical data and clearly defined objective functions. Among them, the Genetic Algorithm (GA) (Holland, 1975), Ant Colony Optimization (ACO) (Dorigo & Gambardella, 1997), and Artificial Bee Colony (ABC) (Karaboga, 2005; Pham et al., 2006) have demonstrated notable effectiveness in solving complex optimization tasks, especially those that are nonlinear, multidimensional, or lack analytical formulations. This paper explores the application of these three metaheuristic algorithms to optimize the weights of criteria within the TOPSIS method, using data obtained from the Vienna Test System applied to a group of candidates for the driving test. Furthermore, two distinct objective functions — the F1-score and the Area Under the Receiver Operating Characteristic Curve (AUC) — are introduced to analyze how different optimization goals affect model performance. A systematic comparison of GA, ACO and ABC efficiency for each objective function is performed, followed by an assessment of the differences between F1-score and AUC-optimized models. The results of this research contribute to enhancing the candidate evaluation process through adaptive weighting and intelligent optimization. By combining TOPSIS with metaheuristic approach, the proposed approach ensures higher accuracy and reliability of assessment, supporting the development of advanced decision-support systems applicable in both civilian and military contexts, where safe and precise vehicle operation is of critical importance.

Problem settings and data representation

The experimental data used in this research originate from a dataset collected through the *Vienna Test System (VTS)*, developed by Schuhfried GmbH. The VTS is a computer-based psychodiagnostic platform designed for standardized assessment of cognitive, psychomotor and perceptual abilities relevant to driving performance. It has been widely implemented in transport, aviation and occupational psychology for the selection and training of drivers and operators in safety-critical environments (Kubinger, 2007; Kaça et al., 2021; Tinella et al., 2021; Masoudi et al., 2022).

In the present study, the VTS was used to evaluate a sample of 583 candidates prior to practical driving examination. Each candidate completed a battery of tests measuring various psychological and motor skills associated with safe vehicle control, such as sustained attention, selective reaction, motor coordination, visual search and decision-making under stress. Table 1 summarizes the main VTS modules used for this study, covering cognitive, psychomotor, and emotional dimensions of performance.

The raw scores obtained from individual tests were normalized to a common scale and aggregated into a matrix $X \in \mathbb{R}^{n \times m}$, where n represents the number of candidates and m the number of considered criteria. Each row of X corresponds to a candidate, and each column to a specific ability index (e.g., reaction time or spatial orientation). For the purpose of evaluation, the target variable y is binary and indicated the final outcome of the driving test (1 – passed, 0 – failed). This setting naturally leads to a multi-criteria classification problem, where the goal is to combine multiple VTS-based criteria into a single composite index of driving ability.

The dataset consisted of percentile scores obtained from the VTS, covering multiple psychometric and cognitive performance criteria for each candidate. All records were complete, with no missing values or invalid entries.

Since VTS tests are expressed in percentiles that differ in scale and dispersion, a z-score normalization was applied to each criterion to ensure comparability and to eliminate potential scale effects in subsequent analysis. This transformation preserved the relative differences between candidates while standardizing all variables to have zero mean and unit variance.

All variables were benefit-type attributes (oriented so that higher values represented better performance), ensuring consistent interpretation across the criteria. The resulting standardized dataset served as an input for evaluation, enabling objective aggregation of psychometric indicators into a single performance score for each candidate.

Table 1 – Classification of criteria in the Vienna Test System

Group	Criterion	Description
COG	COG	Recognition of visual patterns.
DT	DT	Reaction speed to various stimuli (psychomotor + attention).
PP-R	VF	Perception of objects in the peripheral visual field (visual perception).
	TD	Precision of position maintenance and tracking (psychomotor control).
RT	RS	Speed of motor response (psychomotor ability).
	MS	Motor execution speed (psychomotor ability).
ATAVT	ATAVT	Fast recognition of traffic scenes (visual perception).
IVPE	MST	Speed in performing motor tasks (psychomotor speed).
	RSB	Reaction time measured via key press (psychomotor speed).
	SC	Sustained attention and selective reaction to stimuli (attention + concentration).
	RA	Tendency towards risk-taking behavior (cognitive–psychological assessment).
RR	RR	Verbal fluency and attention (attention / verbal fluency).
VIP	SP	Spatial perception and orientation (cognitive ability).
	VIP	Processing and interpretation of visual information (cognitive ability).
	AI	Solving logical and abstract reasoning tasks (cognitive ability).
ED	ED	Decision-making in emotionally challenging situations (emotional and psychological assessment).

TOPSIS with learnable weights

TOPSIS was employed as an MCDM method. The underlying principle of TOPSIS assumes that the best alternative should have the shortest distance from the positive ideal solution (PIS) and the farthest distance from the negative ideal solution (NIS).

In this study, each candidate represents an alternative A_i, while the $m = 16$ criteria derived from the VTS represent quantitative indicators of cognitive and psychomotor performance relevant to driving ability. The decision matrix is defined as:

$$X = [x_{ij}]_{n\times m}, \quad i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, m,$$

where x_{ij} denotes the observed value of criterion C_j for candidate A_i.

Unlike the classical TOPSIS, where weights are predefined or expert-assigned, in learnable weights approach, the weights are optimized using multiple strategies.

This enables the model to *learn* the most informative weighting configuration directly from the data. The resulting weighted normalized matrix, obtained after applying z-score standardization to all criteria, is used to compute the distances to PIS and NIS, yielding a ranking of all candidates. However, unlike traditional MCDM applications, the availability of ground-truth exam outcomes (pass/fail) allows TOPSIS to be evaluated in a classification setting. By comparing the computed preference scores with actual outcomes, we assess how well each weighting strategy discriminates between successful and unsuccessful candidates, using the AUC and the F1-score as performance metrics.

In this study, we focus on the metaheuristic approach of learnable weights within the TOPSIS framework, employing GA, ACO and ABC to discover the most discriminative weighting configurations. Other weighting strategies, including random initialization, regression-based optimization, and fixed (equal-weight) baseline, have been previously examined (Vujadinović et al., 2025).

The pseudocode of the TOPSIS for ranking and optional classification used in this study is given below.

Algorithm 1: TOPSIS for ranking and classification

Input: Criteria matrix X, weight vector w
Output: Ranking of alternatives, success prediction
Normalize the criteria matrix X;
Apply weights w to the criteria;
Determine PIS and NIS;
for *each alternative* **do**
    Compute the distance to PIS and NIS;
    Compute the TOPSIS score as the relative closeness to PIS;
end
Rank the alternatives according to their scores;
Determine the classification threshold τ on validation data;
for *each alternative* **do**
    if *TOPSIS score* $\geq \tau$ **then**
        classify as **successful**;
    end
    else
        classify as **unsuccessful**;
    end
end

Validation procedure

To ensure robustness of the learned weights and to mitigate overfitting to a specific data split, a k-fold cross-validation scheme was employed during optimization. The dataset was partitioned into $k = 5$ disjoint folds, with four folds used for training and one for validation in each iteration.

Each metaheuristic algorithm (GA, ACO, ABC) optimized the weight vector w by maximizing the selected objective function (the AUC or the F1-score) averaged across all folds. The mean validation performance was used as the fitness value guiding the search process.

This procedure ensured that the resulting weights generalize well and that the optimization process reflects the model's stability across different data partitions.

Objective functions

A crucial component of the metaheuristic-based approach process is the *objective function*, which quantifies the quality of each candidate weight vector w during the search. In this study, two alternative objective functions were implemented and the model performance was compared. Each offers a different perspective on model performance — the AUC emphasizes discriminative ability, whereas the F1-score focuses on balanced classification.

AUC as an optimization objective function.

The AUC is a threshold-independent measure of model separability, quantifying how well the continuous TOPSIS scores $C_i(w)$ distinguish between successful and unsuccessful candidates (Hanley & McNeil, 1982; Fawcett, 2006). Mathematically, it can be expressed as:

$$AUC = P(C_+ > C_-),$$

representing the probability that a randomly chosen successful candidate (C_+) has a higher score than a randomly chosen unsuccessful one (C_-). By maximizing the AUC, the optimization process improves overall discriminative power, independent of any specific classification threshold.

F1-score as an optimization objective function.

The F1-score is defined as the harmonic mean of precision and recall, fundamental measures in binary classification problems (Powers, 2011; Sokolova & Lapalme, 2009). Given the confusion matrix components:

- True Positives (TP): correctly predicted successful candidates,
- True Negatives (TN): correctly predicted unsuccessful candidates,
- False Positives (FP): unsuccessful candidates incorrectly predicted as successful, and
- False Negatives (FN): successful candidates incorrectly predicted as unsuccessful.

Precision and recall are computed as:

$$Precision = \frac{TP}{TP + FP}, \quad Recall = \frac{TP}{TP + FN}.$$

The F1-score is then:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall},$$

ranging from 0 (worst) to 1 (best). By maximizing F1, the model seeks to balance false positives and false negatives simultaneously.

When the F1-score is used as the objective function, the metaheuristic algorithm searches for a weight vector that maximizes the average F1-score across validation folds and returns the best solution found during the optimization process. The classification threshold is determined from the Receiver Operating Characteristic (ROC) curve, which represents the trade-off between the True Positive Rate and the False Positive Rate across all possible thresholds. The optimal threshold is selected using Youden's J statistic (Youden, 1950), ensuring the best possible separation between successful and unsuccessful candidates.

AUC-based optimization prioritizes global separability of candidate scores, enabling flexible thresholding for advisory decision-making. F1-score based optimization emphasizes balanced classification performance, minimizing both false approvals and rejections. In this study, both objective functions were applied independently to each metaheuristic (GA, ACO, ABC), yielding complementary insights into model behavior and stability of learned weights.

Each metaheuristic evaluates candidate weight vectors $w^{(k)}$ assigning them the corresponding fitness values. For each vector:

1. Compute TOPSIS scores $C_i(w^{(k)})$.
2. Evaluate the selected objective function (the AUC or the F1-score) on a validation subset.
3. Use the computed metric as the fitness value guiding the search (e.g., selection probability in GA, pheromone deposition in ACO, recruitment probability in ABC).

Optimization continues until convergence or exhaustion of the evaluation budget, with the best-performing weight vector w^* selected for testing on unseen data.

AUC optimized models enhance candidate ranking capability, useful in advisory systems where thresholds may be defined post-hoc. Models optimized for the F1-score reduce both false approvals and rejections, supporting operational decision-making in driver selection.

Metaheuristic

Metaheuristic algorithms represent a broad class of stochastic optimization methods, some of which are inspired by natural processes such as evolution and swarm behavior. They are particularly effective in solving complex, nonlinear and multimodal optimization problems where analytical gradients are unavailable or the search space is discontinuous (Talbi, 2009).

In this study, metaheuristics are used to optimize the TOPSIS weight vector

$$w = (w_1, w_2, \ldots, w_m),$$

subject to the constraints:

$$w_j \geq 0, \quad \sum_{j=1}^{m} w_j = 1.$$

Each algorithm iteratively explores the search space, evaluates candidate weight vectors using the selected objective function (the AUC or the F1-score), and gradually improves them through stochastic operators such as recombination, mutation, and pheromone reinforcement. To ensure a fair comparison, all algorithms were configured with the same computational budget (number of fitness evaluations) and identical stopping criteria.

In this research, three metaheuristic algorithms (GA, ACO and ABC) were used and each of them was adapted to problems characteristics.

Genetic Algorithm (GA)

GA is a population-based stochastic optimization method inspired by biological evolution and Darwinian natural selection.

GA begins with a randomly generated population of weight vectors. Each vector is evaluated using a fitness function $f(w)$, corresponding to the chosen objective function (AUC or F1-score):

$$f(w) = \begin{cases} AUC(w), & \text{if AUC is the optimization objective function,} \\ F1(w), & \text{if F1-score is the optimization objective function.} \end{cases}$$

The evolutionary process consists of iterative application of three GA operators, together with fitness evaluation and a population replacement step:

- *Selection* – parent selection is performed using deterministic elitist rank selection, where the individuals are sorted according to their fitness values (rank-based elitist selection).

- *Crossover* – offsprings are generated using single-point crossover. For each offspring, a random crossover point is selected and the first segment of the weight vector is inherited from one parent, while the remaining segment is inherited from the other parent.
- *Mutation* – for each individual, a fixed percentage of genes is randomly selected and perturbed by adding Gaussian noise. For a selected gene j, the mutated value is given by

$$w_j^{\text{new}} = w_j + \epsilon_j, \qquad \epsilon_j \sim \mathcal{N}(0, \sigma^2).$$

where $\sigma = 0.1$. After mutation, all weights are clipped to the interval $[0, 1]$ and the entire weight vector is normalized to satisfy

$$\sum_{j=1}^{m} w_j^{\text{new}} = 1.$$

Through repeated application of selection, crossover and mutation, the population gradually evolves towards better solutions. The algorithm terminates after a predefined maximum number of generations is reached. The final output is the best weight vector found during the optimization process according to the fitness function.

The pseudocode of the GA implementation used in this study is given below.

Algorithm 2: Genetic Algorithm for optimizing TOPSIS weights

Input: Criteria matrix X, ground truth labels y, GA parameters (max_generations, population size, number of parents, mutation rate)

Output: Best found vector w

Initialize population of random weight vectors
for *generation* $= 1$ **to** *max_generations* **do**
    Evaluate fitness function of each individual using TOPSIS + metric (AUC/F1-score)
    Select the best individuals (elitism + selection)
    Generate offspring using crossover
    Apply mutation to offspring
    Form new population from parents and offspring
end
return *best found vector* w

Ant Colony Optimization (ACO)

ACO is a population-based metaheuristic inspired by the collective foraging behavior of ants, where solution quality is reinforced through pheromone-mediated learning. In the considered formulation, ACO is adapted to a continuous optimization setting in order to determine optimal weight vectors for the TOPSIS method.

In each iteration, a population of n_{ants} ants constructs candidate solutions represented as a continuous weight vector.

The pheromone trail τ_j represents the learned importance of the criterion j and is initialized uniformly with a small random perturbation in order to avoid symmetry and premature bias:

$$\tau_j^{(0)} = 1 + \epsilon_j, \qquad \epsilon_j \sim \mathcal{U}(0, 0.01).$$

Candidate solutions are constructed by combining pheromone-driven exploitation with stochastic exploration of the continuous search space. Specifically, each ant generates a weight vector according to

$$w = \alpha \frac{\boldsymbol{\tau}}{\sum_j \tau_j} + \beta \frac{\boldsymbol{r}}{\sum_j r_j}, \qquad \boldsymbol{r} \sim \mathcal{U}(0,1)^m,$$

followed by an L_1 normalization step, thereby ensuring feasibility of the constructed solution. Here, $\boldsymbol{\tau}$ denotes the pheromone vector, while $\boldsymbol{r}$ is a random vector introducing exploration. The parameters $\alpha \geq 0$ and $\beta \geq 0$ control the balance between pheromone-based exploitation and random exploration. These parameters are kept constant within each ACO run, but are varied across experiments using a predefined parameter grid, with $\alpha \in \{0.8, 1.0\}$ and $\beta \in \{0.1, 0.2\}$.

Each constructed solution is evaluated using a fitness function

$$f(w) \in \{AUC(w),\, F1(w)\},$$

obtained from the TOPSIS-based classifier.

The algorithm employs an *offline pheromone update* mechanism. After all ants construct and evaluate their solutions in a given iteration, pheromone levels are updated in two steps. First, evaporation is applied:

$$\boldsymbol{\tau} \leftarrow (1-\rho)\boldsymbol{\tau},$$

where $\rho \in (0, 1)$ denotes the pheromone evaporation rate. Subsequently, pheromone is reinforced using only a subset of elite solutions from the current iteration. Let $\mathcal{E}$ denote the set of top-performing ants, defined as a fixed fraction of the population. The pheromone update is then given by

$$\boldsymbol{\tau} \leftarrow \boldsymbol{\tau} + \rho \sum_{i \in \mathcal{E}} f(w^{(i)})\, w^{(i)},$$

where $f(w^{(i)})$ denotes the fitness value of the solution $w^{(i)}$ generated in the current iteration.

The algorithm maintains the best found solution across all iterations for reporting purposes. The optimization process is repeated for a fixed number of iterations n_{iter}, which serves as the stopping criterion.

The pseudocode of the ACO implementation used in this study is given below.

Algorithm 3: Ant Colony Optimization for optimizing TOPSIS weights

Input: Criteria matrix X, ground truth labels y, ACO parameters (number_of_ants, max_iterations, α, β, ρ)

Output: Best found vector w

Initialize pheromone vector $\boldsymbol{\tau}$ for all criteria (small random perturbation)
Initialize best found solution w^* and best score
for *iteration* $= 1$ **to** *max_iterations* **do**
    for *each ant* $i = 1, \ldots, number\_of\_ants$ **do**
        Draw random vector $\boldsymbol{r} \sim \mathcal{U}(0,1)^m$ and normalize it
        Construct solution $w^{(i)}$
        Evaluate fitness $f(w^{(i)})$ using TOPSIS + metric (AUC/F1)
        Update (w^*, f^*) if $f(w^{(i)}) > f^*$
    end
    Evaporate pheromone
    Select elite set $\mathcal{E}$ (top n_{elite} ants in the current iteration)
    Offline pheromone update using elite solutions
end
return *best found vector* w

Artificial Bee Colony (ABC)

ABC is a swarm intelligence metaheuristic inspired by the collective foraging behavior of honey bees. The algorithm simulates how bees search for food sources, communicate solution quality, and recruit other bees to promising areas. In ABC, each bee represents a candidate solution encoded as a continuous weight vector.

Each iteration of ABC consists of two complementary phases:

1. *Exploration (scout phase):* scout bees randomly generate new solutions to explore unexplored regions of the search space,
2. *Exploitation (recruitment phase):* non-scout bees follow the best solutions found so far and refine them through local search.

The recruitment probability of the bee i is defined as:

$$P_i = \frac{f(w_i)}{\sum_{k=1}^{N} f(w_k)},$$

where $f(w_i) \in \{AUC(w), F1(w)\}$ denotes the fitness of the weight vector generated by the bee i. Better solutions attract more bees, increasing exploitation pressure on high-quality areas of the search space.

During local exploitation, the weight vector is refined using a stochastic update followed by normalization:

$$w_j = \frac{w_j + \epsilon}{\sum_{k=1}^{m}(w_k + \epsilon)}, \qquad \epsilon \sim \mathcal{N}(0, \sigma^2).$$

The pseudocode of the ABC implementation used in this study is given below.

Algorithm 4: Artificial Bee Colony for optimizing TOPSIS weights

Input: Criteria matrix X, ground truth labels y, ABC parameters (number of bees, max_iterations, ρ, e_frac, sco_str)

Output: Best found vector w

Initialize population of bees with random candidate solutions
for *iteration* $= 1$ **to** *max_iterations* **do**
  Evaluate fitness of all bees using TOPSIS + metric (AUC/F1-score)
  Select elite bees with best solutions
  Redirect non-elite bees towards elite solutions (exploitation)
  Scout bees explore new random solutions (exploration)
end
return *best found vector* w

Results

Experiments were conducted on a dataset consisting of 583 candidates evaluated through VTS tests which includes 16 evaluation criteria (Table 1) and the binary outcome representing pass or fail on the driving exam. All algorithms were implemented in Python and evaluated using 5-fold cross-validation. Each algorithm was executed five times, and the average results were recorded. The performance metrics included Accuracy, the AUC and the F1-score as well as time (s) which represents the internal fitness evaluation time. Precision and Recall were also calculated, although not shown in the tables, since they are required for computing the F1-score. Additionally, execution time and algorithmic complexity were recorded for each optimization approach. The experiments were performed on a standard workstation equipped with an AMD Ryzen 3 processor, 4 GB of RAM, and running Windows 10.

The results of GA metaheuristic

When the optimization objective function is the AUC, GA achieved consistent and interpretable performance across the tested parameter settings. The best results were observed for the configuration of 20 generations (n_gens), 10 individuals (n_pop), 10 parents (n_par), where GA reached an **AUC of 0.7234** and an **accuracy of 0.6585–0.7271**. Increasing the generations number from 10 to 20, the accuracy was improved, indicating that a longer evolutionary process allowed better exploration of the search space and refinement of promising solutions. A mutation rate between 5% and 10% (mut (%)) had a minimal impact on the final AUC, confirming that the algorithm evolved in a stable manner and was not overly sensitive to stochastic variations.

In some configurations, overall accuracy increased while the AUC slightly decreased. This indicates that although the model correctly classified a higher proportion of samples, its ability to consistently rank positive versus negative cases was slightly reduced. Such behavior reflects the inherent difference between the accuracy and the AUC as evaluation metrics.

Table 2 – Best 5 GA results for objective functions AUC and F1-score

Obj	n_gens	n_pop	n_par	mut (%)	Accuracy	AUC/F1	time (s)
AUC	20	10	10	5	0.6585	0.7234	1.79
AUC	20	10	10	10	0.6585	0.7234	1.96
AUC	20	20	10	5	0.7271	0.7233	3.82
AUC	20	20	10	10	0.7271	0.7233	3.66
AUC	30	20	10	5	0.7308	0.7227	5.78
F1	30	10	5	5	0.7737	0.8468	2.82
F1	30	10	5	10	0.7737	0.8468	2.84
F1	20	10	5	5	0.7326	0.8039	1.94
F1	20	10	5	10	0.7326	0.8039	1.91
F1	20	20	10	5	0.7017	0.7765	3.76

When the optimization objective function was shifted to the F1-score, the GA adapted its search behavior accordingly. The top-performing configuration 30 generations, 10 individuals, 5 parents, 5% mutation achieved an

F1-score of 0.8468 with an **accuracy of 0.7737**. Unlike the AUC-based runs, no trade-off between accuracy and the F1-score was observed in this case. Other configurations with slightly different mutation rates (5–10%) yielded identical performance, reflecting stable convergence behavior. Overall, GA demonstrated high robustness and adaptability to different objective function formulations, maintaining solid performance and minimal sensitivity to parameter tuning.

Table 2 presents the five best-performing results for both objective functions, the AUC and the F1-score.

The results of ACO metaheuristic

When optimized for the AUC-based fitness function, the ACO algorithm demonstrated consistent convergence and solid discriminative performance. The best configuration was achieved with $\alpha = 1.0$, $\beta = 0.1$, $\rho = 0.2$ and a colony size of 20 ants (n_ants) and 50 iterations (n_iter). This setup resulted in an **AUC of 0.7528** with an **accuracy of 0.7479**. The relatively high heuristic importance (β) guided ants towards more promising regions in the search space, while a moderate pheromone evaporation rate (ρ) maintained useful historical information without premature stagnation. The algorithm achieved a strong balance between exploration and exploitation, as evidenced by a stable convergence curve and low variance across folds. Overall, ACO optimized for the AUC favored global ranking consistency and maintained well-balanced classification metrics across classes.

Table 3 – Best 5 ACO results for objective functions AUC and F1-score

Obj	n_ants	n_iter	ρ	α	β	Accuracy	AUC/F1	time (s)
AUC	20	50	0.2	1.0	0.1	0.7479	0.7528	9.39
AUC	40	50	0.1	0.8	0.1	0.7393	0.7527	17.94
AUC	20	50	0.2	0.8	0.1	0.7565	0.7525	9.12
AUC	20	50	0.1	1.0	0.2	0.7065	0.7519	9.19
AUC	40	100	0.1	1.0	0.1	0.7530	0.7519	35.87
F1	20	100	0.1	1.0	0.2	0.7134	0.7780	18.02
F1	40	100	0.1	0.8	0.1	0.7168	0.7699	36.48
F1	40	50	0.2	1.0	0.2	0.7098	0.7640	18.17
F1	20	50	0.2	0.8	0.1	0.7081	0.7640	9.35
F1	20	100	0.2	0.8	0.2	0.7064	0.7638	18.24

When the F1-score is used as the objective function, ACO shifts its search towards solutions that prioritize the correct identification of successful candidates. This leads to a higher **F1-score of 0.7780** obtained at $\alpha = 1.0$, $\beta = 0.2$, $\rho = 0.1$, 20 ants and 100 iterations but with a noticeably lower **accuracy of 0.7134**. This indicates that F1-based optimization pushes ACO to favor sensitivity towards the majority class, even at the cost of overall classification balance. Compared to AUC optimization, the F1-score optimized configurations show greater variability, meaning the search space is less smooth and more sensitive to hyperparameter settings.

The five best results obtained using the AUC and the F1-score as objective functions are shown in Table 3.

The results of ABC metaheuristic

In the AUC-optimized mode, the ABC algorithm achieved its best performance with a population of 20 bees (n_bees), 50 iterations (n_iter), an evaporation rate of $\rho = 0.2$, elite fraction of 0.2 (e_frac), and scout strength (sco_str) of 0.05. This configuration produced an **AUC of 0.7455** with an **accuracy of 0.7598**. The balance between exploration and exploitation was effectively maintained through moderate elitism and controlled scout activity, enabling the algorithm to refine high-quality solutions without excessive random wandering. The strong AUC performance indicates that ABC successfully ranked candidates in alignment with the target classification boundaries. Stability across multiple runs confirms that these parameter values provide a robust trade-off between convergence speed and solution quality.

Table 4 – Best 5 ABC results for objective functions AUC and F1-score

Obj	n_bees	n_iter	ρ	e_frac	sco_str	Accuracy	AUC/F1	time (s)
AUC	20	50	0.2	0.2	0.05	0.7598	0.7455	8.99
AUC	20	100	0.1	0.2	0.1	0.7271	0.7453	18.52
AUC	40	50	0.2	0.2	0.05	0.7340	0.7445	18.43
AUC	40	100	0.2	0.2	0.1	0.7340	0.7445	36.29
AUC	20	100	0.2	0.1	0.05	0.7597	0.7445	18.41
F1	20	50	0.1	0.1	0.1	0.7598	0.8346	8.99
F1	20	50	0.1	0.2	0.1	0.7547	0.8316	9.01
F1	40	50	0.2	0.1	0.1	0.7410	0.8149	18.14
F1	20	50	0.1	0.1	0.05	0.7204	0.7861	9.09
F1	20	50	0.2	0.1	0.1	0.7102	0.7854	9.20

When optimizing for the F1-score, ABC exhibited more aggressive exploitation behavior, focusing on reducing false negatives. The best-performing setup used 20 bees, 50 iterations, an evaporation rate of $\rho =$

0.1, elite fraction of 0.1, and scout strength of 0.1, resulting in an **F1-score of 0.8346** with an **accuracy of 0.7598**. Compared to the AUC-oriented version, this configuration converged faster but with higher variability across runs. The increased scout activity improved the exploration of new candidate regions, yet introduced minor instability in fitness progression. Despite this, the F1-score based optimization was particularly effective in identifying successful candidates, achieving strong class sensitivity performance and competitive classification accuracy.

Table 4 presents the five best-performing results for both objective functions, the AUC and the F1-score.

Comparison between the AUC and F1-score objective functions

A direct comparison between the two optimization objective functions reveals that the AUC-based optimization yields more stable and generalizable models, while the F1-score based optimization prioritizes performance on the dominant class. The difference in overall accuracy was minor, yet the class-wise behavior diverged significantly: The AUC optimization maintained better overall class balance, whereas the F1-score optimization improved detection of successful candidates but at the cost of false negatives. This highlights the importance of selecting an appropriate objective function according to the intended application—whether balanced discrimination or maximal recognition of a specific class is desired.

When the AUC is used as the optimization objective function, all three metaheuristic algorithms (GA, ACO, ABC) tend to converge towards weight vectors that improve the global discriminative power of the TOPSIS model. In this setting, the algorithms search for solutions that maximize the separation between successful and unsuccessful candidates across the entire scoring scale, rather than focusing on the performance of a specific class. As a result, the AUC-based optimization produces more balanced and stable models, with relatively low variability between executions and smoother convergence. This behavior is particularly suited for ranking problems, where the goal is to ensure that higher-ranked candidates are consistently better than lower-ranked ones according to all criteria.

In contrast, when the objective function is the F1-score, the optimization becomes strongly oriented towards correct identification of the positive class, i.e., candidates who successfully pass the driving exam. The search focuses on solutions that increase sensitivity and improve detection of successful candidates, even if this leads to slightly lower overall ranking precision or weaker class separation. In this mode, the metaheuristic tends to emphasize exploitation of the search space, producing higher F1-score values but also greater variability across parameter configurations. This makes F1-score optimized solutions particularly suitable when false negatives (misclassifying successful candidates as unsuccessful) are considered more costly than false positives.

In summary, the AUC-oriented optimization yields stable, globally discriminative models suitable for ranking, whereas the F1-oriented optimization yields sensitive, class-focused models suitable for decision-making and risk detection. The choice of the objective function should therefore follow the practical requirements of the evaluation system: whether the goal is to build an accurate ranking of candidates or to minimize the risk of approving an unfit driver.

Time-budget comparison of metaheuristic-based weight optimization methods

To enable a fair comparison between metaheuristics with different convergence speeds, all algorithms are evaluated under the same fixed time budget of 300 seconds, using their previously optimized parameter settings.

Table 5 summarizes the best-found results obtained by GA, ACO, and ABC under a fixed time budget. The reported best-target value represents the best average cross-validation performance obtained within the allotted time budget.

Table 5 – Best-found performance of GA, ACO, and ABC under a fixed time budget

Variant	Target	Best target	Precision	Recall	F1	AUC
GA	AUC	0.748	0.896	0.661	0.758	0.748
ACO	AUC	0.765	0.898	0.740	0.802	0.765
BCO	AUC	0.761	0.903	0.746	0.805	0.761
GA	F1	0.837	0.853	0.828	0.837	0.692
ACO	F1	0.807	0.900	0.740	0.807	0.743
BCO	F1	0.800	0.878	0.747	0.800	0.713

Under these conditions, the evaluated metaheuristics exhibit distinct and complementary strengths depending on the optimization objective function. When the AUC is used as the objective function, ACO and ABC consistently achieve higher best-found AUC values than GA, indicating a stronger ability to optimize the global ranking of candidates across the entire score distribution. This behavior can be attributed to pheromone-based information sharing, which promotes collective exploration of promising regions and leads to smoother and more stable ranking structures. In contrast, when the F1-score is directly optimized, GA attains the highest F1-score values, reflecting its effectiveness in fine-grained adjustment of weight configurations that balance precision and recall at a specific decision threshold. The evolutionary operators of GA, particularly mutation and elitist selection, enable more aggressive local refinements that are beneficial for threshold-dependent metrics. As a result, improvements in the F1-score are achieved even when global ranking quality, as measured by the AUC, is not maximized. These findings confirm that no single metaheuristic dominates across objective functions and emphasize that the selection of the optimization algorithm should be aligned with the intended evaluation criterion and decision-making requirements.

Computational complexity

The computational cost of the proposed approach consists of two parts: (1) evaluation of the TOPSIS method for a given weight vector, and (2) iterative optimization performed by a metaheuristic algorithm (GA, ACO or ABC).

For a decision matrix $X \in \mathbb{R}^{n \times m}$ containing n candidates and m criteria, one evaluation of TOPSIS requires: normalization of the matrix, weighting of the criteria, computing distances to PIS and NIS and calculating the closeness coefficient. This results in a computational complexity of:

$$\mathcal{O}(nm + n \log n),$$

where the first term corresponds to matrix operations and the second term to sorting performed during evaluation of performance metrics.

Each metaheuristic performs a number of TOPSIS evaluations. If E denotes the number of candidate solutions generated during optimization, then the total complexity of the optimization process is:

$$\mathcal{O}(E \cdot (nm + n \log n)).$$

Since E depends only on the algorithm parameters (population size, number of generations/iterations) and not on the data itself, the complexity grows linearly with the number of fitness evaluations. Therefore, all three algorithms (GA, ACO, ABC) exhibit the same asymptotic behavior, differing only in the value of E.

In practice, the execution time is dominated by the number of fitness evaluations, since each evaluation requires running TOPSIS and computing classification metrics. Because TOPSIS operates on relatively small matrices (a fixed number of criteria), the metaheuristic approach remains computationally efficient and can be executed on a standard workstation without parallelization.

Although all three metaheuristic algorithms operate under the same asymptotic computational complexity, their execution time differs in practice due to the way each algorithm generates and evaluates candidate solutions.

GA maintains a fixed-size population and produces a limited number of new solutions per generation through crossover and mutation operators. Consequently, the number of fitness evaluations E grows proportionally to the number of generations and the population size. Since GA typically converges quickly towards promising regions of the search space, fewer candidate weights are evaluated overall.

In contrast, ACO and ABC are based on swarm-search mechanisms, where a significantly larger number of solutions is generated during each iteration. In ACO, every ant probabilistically constructs a new solution using pheromone trails, which results in many more candidate weight vectors evaluated in each step. Similarly, in ABC, both recruited bees and scout bees explore the search space simultaneously, leading to multiple parallel solution updates per iteration.

Because each newly generated solution requires a complete execution of TOPSIS, followed by evaluation of classification metrics (the F1-score or the AUC), the total execution time increases proportionally to the number of solutions explored. Therefore, ACO and ABC require more computational time than GA, not because their per-iteration complexity is higher, but because they evaluate a substantially larger number of candidate weight vectors during the optimization process.

Sarić, I. et al., Metaheuristic-based approach to optimizing the weights in the TOPSIS method for driver candidate performance assessment, pp.1196–1221

Conclusion

This paper presented a metaheuristic-based optimization framework for improving the weight determination process in the TOPSIS method when evaluating the performance of driver candidates based on psychomotor and cognitive abilities measured by the VTS. Unlike traditional TOPSIS implementations that rely on fixed or subjectively assigned weights, the proposed approach automatically learns the optimal weight configuration by using three population-based optimization algorithms: GA, ACO, and ABC.

Two optimization objective functions were analyzed — the AUC and the F1-score. The AUC encouraged the algorithm to improve the overall separability of candidates, resulting in stable and balanced ranking performance. In contrast, the F1-score optimization directed the search towards maximizing correct identification of successful candidates, emphasizing sensitivity of the classification process. This makes the F1-score an objective function particularly suitable when the decision system prioritizes reducing classification errors in determining whether a candidate is ready to proceed to practical driving examination.

Among the evaluated metaheuristics, ACO achieved the highest mean AUC value, indicating superior capability in separating successful from unsuccessful candidates, whereas ABC reached the highest mean F1-score. GA demonstrated competitive performance with a significantly lower computational cost, making it suitable for applications with limited execution time.

The findings confirm that introducing the metaheuristic approach into TOPSIS substantially improves decision accuracy and removes the need for subjective weight assignment. This contributes to the development of data-driven, objective function, and scalable evaluation tools for candidate assessment in transportation safety domains. Future research may extend this work by incorporating additional machine learning models, testing hybrid optimization strategies, or validating the approach on larger datasets and real-world licensing processes.

References

Dorigo, M. & Gambardella, L.M. 1997. Ant colony system: a cooperative learning approach to the traveling salesman problem, *IEEE Transactions on Evolution-*

ary Computation, 1(1), pp. 53–66. Available at: https://doi.org/10.1109/4235.585892

Fawcett, T. 2006. An introduction to ROC analysis, Available at: https://doi.org/10.1016/j.patrec.2005.10.010

Hanley, J.A. & McNeil, B.J. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve, *Radiology*, 143(1), pp. 29–36. Available at: https://doi.org/10.1148/radiology.143.1.7063747

Holland, J.H. 1975. Adaptation in Natural and Artificial Systems, *Ann Arbor: University of Michigan Press.*

Huang, S., Guo, Y., Zhang, X. & Li, J. 2020. Comprehensive evaluation of expressway safety based on AHP–TOPSIS–RSR. *Physica A: Statistical Mechanics and its Applications*, 551, 124–134.

Hwang, C.L. & Yoon, K. 1981. *Multiple Attribute Decision Making: Methods and Applications*. Berlin: Springer.

Kaça, G., Izmitligil, T., Koyuncu, M. & Amado, S. 2021. How well do the traffic psychological assessment systems predict on-road driving behaviour?, *Applied Cognitive Psychology*, 35(5), pp.1321–1337. Available at: https://doi.org/10.1002/acp.3867

Karaboga, D. 2005. An idea based on honey bee swarm for numerical optimization, *Technical Report TR06*, Erciyes University, Turkey.

Kubinger, K.D. 2007. Psychological test calibration using the Rasch model: some critical remarks. *Psychology Science*, 49(1), pp.49–56. Available at: https://doi.org/10.1207/s15327574ijt0504_3

Masoudi, N., Rezaei, M., Farahbakhsh, M., Zamani-Sani, H., Abdi, S. & Sadeghi-Bazargani, H. 2022. Evaluating the risk of road traffic accidents (RTAs) and the reaction time of individuals with attention deficit disorder (ADD) in the virtual environment: A study protocol using the Sahand driving simulator and the Vienna test system, *Medical Journal of Tabriz University of Medical Sciences*, 44(5), pp.367–379. Available at: https://doi.org/10.34172/mj.2022.041

Pham, D.T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S. & Zaidi, M. 2006. The bees algorithm — A novel tool for complex optimization problems, in: *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems (IPROMS 2006)*, pp. 454–459. Available at: https://doi.org/10.1016/B978-008045157-2/50081-X

Powers, D.M.W. 2011. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation, *Journal of Machine Learning Technologies*, 2(1), pp. 37–63. Available at: https://doi.org/10.9735/2229-3981

Ren, W., Zhang, X. & Zhao, M. (2021). A comprehensive traffic risk assessment model for urban roads based on entropy–TOPSIS. *Safety Science*, 138, 105236.

Schuhfried GmbH. 2013. *Vienna Test System: User Manual*. Mödling, Austria: Schuhfried GmbH. Available at: https://www.schuhfried.com/en/

Sokolova, M. & Lapalme, G. 2009. A systematic analysis of performance measures for classification tasks, *Information Processing & Management*, 45(4), pp. 427–437. Available at: https://doi.org/10.1016/j.ipm.2009.03.002

Talbi, E.G. 2009. Metaheuristics: From Design to Implementation, *Hoboken*, New Jersey: Wiley. Available at: https://www.wiley.com/

Tinella, L., Caffò, A.O., Lopez, A., Nardulli, F., Grattagliano, I. & Bosco, A. 2021. Reassessing fitness-to-drive in drinker drivers: The role of cognition and personality, *International Journal of Environmental Research and Public Health*, 18(23), 12828. Available at: https://doi.org/10.3390/ijerph182312828

Vujadinović, J., Sarić, I. & Nikolić, R. 2025. Assessment of candidates performance in the driving test using the TOPSIS method and its modifications, in: *Proceedings of the International Symposium on Operational Research – SIMOPIS 2025*, Palić, 7–10. September 2025, pp. 225–230 (in Serbian). Available at: https://www.symopis2025.fon.bg.ac.rs/

Youden, W.J. 1950. Index for rating diagnostic tests, *Cancer*, 3(1), pp. 32–35. Available at: https://doi.org/10.1002/1097-0142(1950)3:1<32::aid-cncr2820030106>3.0.co;2-3

Метахеуристички приступ оптимизацији тежина у методи TOPSIS за процену успеха кандидата на возачком испиту

Ивана М. Сарић[а], *Јасмина* Ц. Вујадиновић[б], *Рале* М. Николић[в]

[а] Универзитет у Београду, Технолошко-металуршки Факултет, Катедра за Математичке науке, Београд, Република Србија

[б] Универзитет одбране у Београду, Војна академија, Катедра Природно-математичких наука, Београд, Република Србија
аутор за преписку

[в] Универзитет одбране у Београду, Војна академија, Катедра Природно-математичких наука, Београд, Република Србија

ОБЛАСТ: математика, вишекритеријумско одлучивање, когнитивна процена у евалуацији кандидата за возача
КАТЕГОРИЈА (ТИП) ЧЛАНКА: оригинални научни рад

Сажетак:

Увод/циљ: Безбедност у саобраћају и поуздана селекција возача представљају важан сегмент савременог друштва. Циљ овог рада је унапређење поступка процене успешности кандидата на возачком испиту применом вишекритеријумских метода одлучивања и метахеуристичке оптимизације. На основу резултата добијених тестом Vienna Test System, предложена је примена методе TOPSIS са адаптивним одређивањем тежина критеријума.

Методе: Тежински коефицијенти методе TOPSIS оптимизовани су помоћу три метахеуристичка алгоритма – генетског алгоритма (GA), алгоритма мрављих колонија (ACO) и алгоритма пчелињих колонија (ABC). Током оптимизације коришћене су две различите функције циља: AUC и F1-score, како би се испитало њихово дејство на тачност и стабилност модела. Експериментални оквир обухвата три сегмента: (1) поређење перформанси GA, ACO и ABC метахеуристика за AUC функцију циља, (2) аналогно поређење за F1-score функцију циља и (3) међусобну анализу AUC и F1-score оптимизованих модела.

Резултати: Указано је да избор метахеуристичког алгоритма и функције циља знатно утичу на перформансе методе TOPSIS. Оптимизација са AUC функцијом циља довела је до стабилнијих модела и бољег баланса између успешних и неуспешних кандидата, док је оптимизација са F1-score функцијом циља постигла већу осетљивост и бољу идентификацију успешних кандидата.

Закључак: Увођење метахеуристичких алгоритама у оптимизацију тежина методе TOPSIS омогућава адаптивно и поузданије рангирање кандидата, чиме се доприноси развоју интелигентних система за селекцију возача и унапређењу безбедности у саобраћају. Добијени резултати потврђују да се правилним избором функције циља и алгоритма оптимизације може постићи знатно побољшање тачности модела.

Кључне речи: TOPSIS, метахеуристике, GA, ACO, ABC, F1-score, AUC, Vienna Test System, вишекритеријумско одлучивање.

Paper received on: 13.11.2025.
Manuscript corrections submitted on: 05.12.2025.
Paper accepted for publishing on: 26.12.2025.