

ŽELJKO KANOVIĆ, Ph.D.

E-mail: kanovic@uns.ac.rs

VLADIMIR BUGARSKI

Computing and Control Department

TODOR BAČKALIĆ, Ph.D.

E-mail: tosa@uns.ac.rs

Department for Traffic Engineering

Faculty of Technical Sciences, University of Novi Sad

Trg Dositeja Obradovića 6, 21000 Novi Sad, Serbia

Swarm Intelligence in Transportation Engineering

Preliminary Communication

Accepted: May 20, 2013

Approved: Feb. 12, 2014

## SHIP LOCK CONTROL SYSTEM OPTIMIZATION USING GA, PSO AND ABC: A COMPARATIVE REVIEW

### ABSTRACT

*This paper presents the comparison of some well-known global optimization techniques in optimization of an expert system controlling a ship locking process. The purpose of the comparison is to find the best algorithm for optimization of membership function parameters of fuzzy expert system for the ship lock control. Optimization was conducted in order to achieve better results in local distribution of ship arrivals, i.e. shorter waiting times for ships and less empty lockages. Particle swarm optimization, artificial bee colony optimization and genetic algorithm were compared. The results shown in this paper confirmed that all these procedures show similar results and provide overall improvement of ship lock operation performance, which speaks in favour of their application in similar transportation problem optimization.*

### KEY WORDS

*ship lock, fuzzy expert system, particle swarm optimization, artificial bee colony optimization, genetic algorithm*

### 1. INTRODUCTION

Ship locks are the oldest and the most often used waterway structures that enable vessels to surmount water level differences in a simple and safe way. Dam construction presents an artificial obstacle and creates differences in water level. Ship locks are designed to enable ships to overcome rises in the water level and help to maintain navigation on inland waterways [1]. A ship lock or navigation lock is a hydraulic structure that consists of an enclosed chamber with watertight gates at each end. The water level difference is surmounted by filling or emptying the lock chamber. In other words, lock operates on the simple buoyancy principle that any vessel, no matter what size, will float

atop a large enough volume of water. By raising or lowering the level of a body of water, the vessel itself goes up or down accordingly. The ship lock operators or lock masters always attempt to fill or empty the lock in the fastest time possible with a minimum of turbulence. The organization of vessel traffic on a waterway in the zone of a ship lock is a compromise between rational utilization of the lock and minimizing of ship's delay while waiting to transit the lock [2-4]. The basic elements of a classic ship lock are presented in [2, 5].

This paper presents a Fuzzy Expert System (FES) designed to assist the ship lock operators in the decision-making process. From a wide range of types of ship locks the choice was narrowed to a system that is usually applied on navigable channels on inland waterways: single-channel queuing system with two independent, stochastic streams of arrivals from two opposite directions. Although the model has been established and tested in a particular real system, the principle of generality is not lost. With minor changes in the design of FES, the proposed model can be extended to any other lock from the observed category.

Campbell [6] presented the decision tools for reducing congestion at locks on the upper Mississippi river. Bugarski, Bačkalić and Kuzmanov [5] proposed a fuzzy decision support system for controlling a ship lock. Fuzzy logic is chosen as a control method that does not require a precise mathematical model of the controlled system [7] and as the most suitable mathematical approach for addressing uncertainty, subjectivity, polysemy and indefiniteness [8]. Other authors [9, 10] have also used fuzzy reasoning in decision-support processes. Teodorović and Vukadinović [11] successfully applied fuzzy logic and artificial intelligence in traffic control. Recent applications of fuzzy expert systems may be found in [5, 12, 13].

The main objective of this study is to optimize the performance of fuzzy expert system controlling the ship lock, in order to achieve the best value of the economic criterion defined as a linear combination of two opposite criteria. The first one is a minimum number of empty lockages (lockages without a vessel), and the second one is minimal waiting time (ship's delay). Fuzzy system is a complicated decision system, described by highly non-linear and logic functions, and it is very difficult to obtain a model of such a system described by analytical expressions. Even if we could obtain an accurate analytical model, it would still be a non-convex optimization problem with more local optimal solutions. Thus, it is chosen to apply global numerical optimization algorithms, which provide thorough investigation of the search space and, in this particular case, a more reliable solution than some classic analytical approaches. Three popular global optimization algorithms were used: Particle Swarm Optimization (PSO), Artificial Bee Colony Optimization (ABC) and Genetic Algorithm (GA), with objective to find the best optimization technique for the presented expert system controlling a ship lock process. All these algorithms have been frequently used in engineering applications [14-17]. The results obtained in this research proved that it is possible to design an optimal FES enabling control over economic performance of the entire system, and also that global optimization algorithms used in the study can be successfully applied in problems concerning transportation performance improvement and optimization.

The paper is organized as follows: Section 2 describes the lock parameters and research data. Section 3 discusses the proposed methodology and the process of optimization. In Section 4, the empirical results are summarized and discussed. Finally, Section 5 contains the concluding remarks.

## 2. RESEARCH DATA

The ship lock "Kucura" (Figure 1) on the Danube-Tisa-Danube hydro system in Serbia was observed as a representative real system. Time intervals for the lockage (i.e., passage) were defined as result of time measurement and interview with lock operators. The average time for the lockage of a vessel is adopted to be 25 minutes, and the time interval for the change of level in the chamber without vessel - 15 minutes. Furthermore, it is assumed that the lock chamber can hold only a single vessel. Based on these time intervals, the "regular lockage" (when the vessel enters the end of a lock where the gate is open and does not have to wait for the water level to change in the chamber) takes 25 minutes. The "empty lockage" is the situation when the vessel approaches from the end where the lock gate is closed, and before lockage can take place,

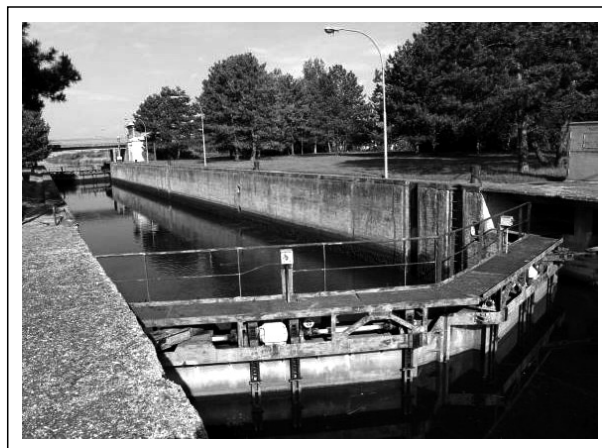


Figure 1 - Ship lock "Kucura"

the water level in the chamber must be changed. An empty lockage followed by the regular lockage takes 40 minutes.

For every new arrival of a vessel, the control logic must decide whether to perform regular or empty lockage. Depending on that decision, the number of empty lockages and the total waiting time are increased by algorithm presented in Figure 2. At the very end of the simulation average waiting time is calculated dividing the total waiting time with the total number of vessels.

A compromise between minimizing the waiting time for the lockage and minimizing the energy and water consumption for operating the lock is the main objective in the ship lock control problem [18]. The owners of the lock prefer fewer empty lockages because such lockages reduce the operating costs. However, ship-owners prefer to increase commercial speed of ships, i.e. wait as little as possible for the transition. In the case when several vessels are approaching the lock from the same direction, the lock operators have to change the level of the water in the empty chamber in order to reduce the waiting times, and this increases the costs of lock operation.

Set of ship arrivals was generated using cumulative statistic data on ship arrivals, which was the only available data. Some stochastic parameters were then applied to generate time schedule of ship arrivals used in the simulation. This set can be considered as a ship traffic database. In the observed case, there is an annual cessation of the navigation during winter (from the 21<sup>st</sup> of December to the 21<sup>st</sup> of March), which is included in the construction of the set of arrivals. On other days, the traffic load is approximately 10 ships per day. There are a total of 2,786 generated arrivals at the lock (presented by month in Table 1).

### 2.1 Fuzzy expert system

The basic fuzzy expert system was constructed by interviewing the lock masters and it is presented in

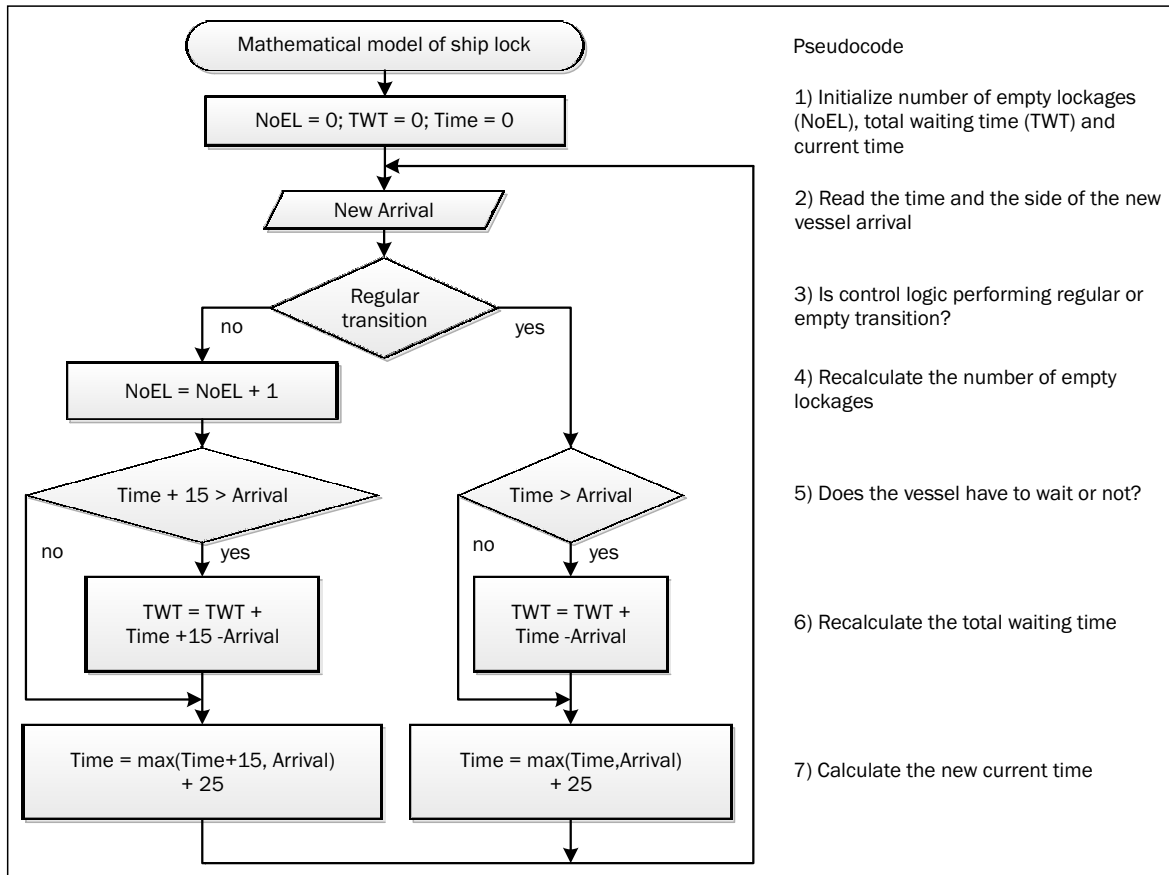


Figure 2 - Algorithm and pseudocode of the number of empty lockages (NoEL) and total waiting time (TWT) calculation

Table 1 - Summary of ship arrivals per month generated for simulation

Month	Total number of ships	Arrivals at upper gate of lock	Arrivals at lower gate of lock	Ratio of arrivals up/down
March	106	50	56	1.12
April	311	147	164	1.11
May	322	167	155	0.93
June	306	161	145	0.90
July	289	143	146	1.02
August	313	161	152	0.94
September	297	153	144	0.94
October	294	166	128	0.77
November	325	152	173	1.14
December	223	118	105	0.89

[5]. Regarding the guidelines and the recommendations of the Inland Navigation Commission [19], FES is designed to be compatible with the River Information Service (RIS). FES collects information from RIS such as data on speed, distance and the direction from which the ship is coming [20].

Based on the state of the lock (lower or upper gate open), FES considers two variables: the distance of the ship from the lock on the level where the gate is open (LGO) and the distance of the ship from the lock on the level where the gate is closed (LGC). The FES decides

whether to change the present state of the ship lock according to the ship distances at both levels.

Three categories related to the distance from the ship lock (small, medium and large) define fuzzy sets for two fuzzy input variables, LGO (Figure 3) and LGC (Figure 4). The output variable represents the control variable “change of condition of the lock” (LC), which is expressed in three categories: change, no change and indefinite (Figure 5). Distances from the lock (input variables) are expressed in minutes, and output value after defuzzification is given in universal units

and after comparison to the limit value gives a binary decision (“change” or “no change”). Table 2 presents the nine fuzzy rules.

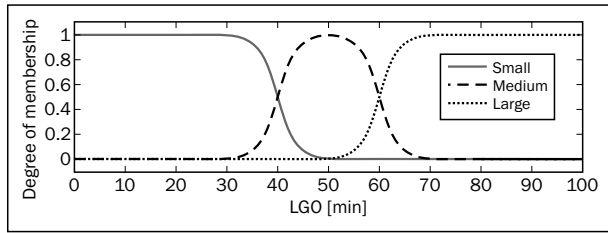


Figure 3 - Membership functions of input fuzzy variable LGO (distance of a ship from the ship lock on the level where the gate is open)

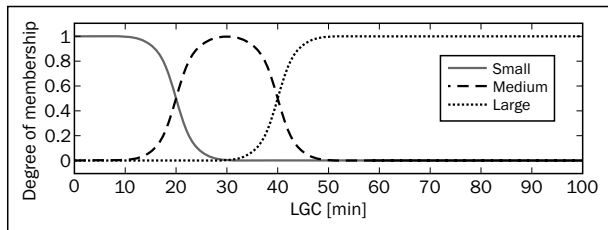


Figure 4 - Membership functions of input fuzzy variable LGC (distance of a ship from the ship lock on the level where the gate is closed)

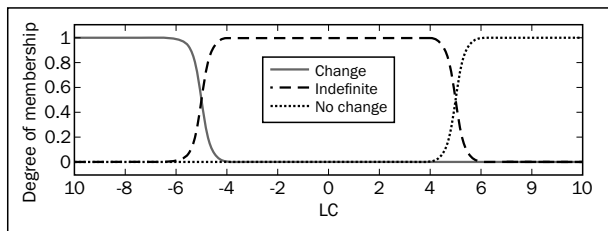


Figure 5 - Membership functions of output fuzzy variable LC (change of condition of the lock)

The relation between the membership function of the fuzzy set and the observed variable is described by a logistic curve (e.g., S-curve or sigmoid function), as shown by several authors on the basis of experiments and research [3, 21-23]. The logistic curve has the mathematical form:

$$\mu(x) = \frac{1}{1 + e^{-a(x-b)}} \quad (1)$$

where:

- a - slope of the function;
- b - point of function gradation (inflection point of function).

Table 2 - Fuzzy rules

LGO	LGC		
	Small	Medium	Large
Small	No change	No change	No change
Medium	Indefinite	No change	No change
Large	Change	Indefinite	No change

The fuzzy decision mechanism is a process consisting of three phases: implication, aggregation and defuzzification [24-26]. The choice of the functions that will implement these three phases has a significant impact on the results and the algorithm speed [27]. That is why most of the practical implementations of fuzzy logic are based on Takagi-Sugeno type, but in our case speed of the algorithm is not important because of slow nature of the ship locking process. Thus, in this study the basic Mamdani type FES (minimization for implication, maximization for aggregation and Centre of Area for defuzzification) was used. This combination of methods is a very common one [26, 27]. However, other combinations of methods are taken into consideration and proposed combination gave the best results in all of our test cases.

### 3. OPTIMIZATION OF MEMBERSHIP FUNCTIONS

As mentioned earlier, in the operation of the ship lock, there are opposing interests of shippers and lock owners. Based on these two interests, it is possible to construct two extremely opposite criteria: Minimum waiting time (MWT) and Minimum number of lockages (MNL). The main concern of shippers is that the lockage is completed as soon as possible; i.e., that they spend the least amount of time waiting in the lock zone. The most convenient situation for them is when the lockage starts immediately upon arrival of the ship. The first criterion reflects precisely that goal - MWT. MNL represents the interests of lock owners and workers. Any change in the water level in the chamber without a vessel inside is an unnecessary expenditure of energy (to run the pumps) and water (which fills the chamber). When using the FES in ship lock control, there is a compromise, because it has efficiency between those two extreme criteria. Because the FES was built based on expert suggestions, it can be concluded that the operator performance in ship lockage would be near to FES performance. When the actions of the FES were analyzed, some disadvantages of this approach were noted. The majority of lockages resulted in the activation of only two of the nine rules. Since this case was not rare, we considered further adjustments of the control logic. If we can design one overall criterion for evaluating the ship lock control, then we can optimize our FES to give the optimal performance for this criterion. Since we have two objectives, this must be a multi-objective optimization [28, 29].

Fuzzy rules were created based on subjective descriptions of lock master and are presented in Table 2. The work principle of fuzzy inference is such that it mimics the human reasoning. However, the question is whether the obtained fuzzy sets are the best choice for the quality control of the ship lock. Can computer



find better control tactic than a human can? Is it possible, with some changes in membership functions, to improve the obtained results?

In order to compare the work of the various fuzzy expert systems it is necessary to form an assessment, i.e. a universal criterion. In this case, such a criterion can be conceived as an “economic” criterion. This criterion is actually a weighted sum of the number of empty lockages and the average waiting time per vessel (see Equation 2).

$$E = A * NoEL + B * AWTpS \quad (2)$$

where:

$E$  – optimization criterion,

$A, B$  – weight coefficients,

$NoEL$  – number of empty lockages,

$AWTpS$  – average waiting time per ship.

Coefficients  $A$  and  $B$  give greater or lesser importance to each of the two components of the economic criterion. With the relation between them we can choose what is more expensive, the waiting of the ships or the waste of water and energy.

Different values of coefficients  $A$  and  $B$  are used to form several optimality criteria. Three well-known and popular optimization techniques will be used: genetic algorithm (GA), particle swarm optimization (PSO), and artificial bee colony optimization (ABC). All three algorithms belong to the group of evolutionary optimization algorithms. PSO and ABC are also classified as members of subgroup named swarm intelligence. A brief description of these algorithms will be presented in the sequel.

### 3.1 Genetic algorithm

Genetic algorithm is an evolutionary optimization technique inspired by Darwin’s theory of natural evolution of the species. It was proposed in 1970s by John Holland [30] and improved during the years by numerous other researchers [31]. In this technique, a population of candidate solutions (called individuals) to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered.

The evolution usually starts from a population of randomly generated individuals and is an iterative process. The population in each iteration is called a generation. The value of the objective function, called fitness, of every individual in the population is evaluated in each generation. The fitter individuals are selected from the current population, and each individual’s genome is modified (recombined and possibly randomly mutated) to form a new generation. The new generation of candidate solutions is then used in the next iteration of the algorithm. When forming a new population, the evolution mechanisms are used, such

as selection, crossover and mutation, imitating the process of natural evolution. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population.

Since genetic algorithms are frequently applied in many application areas, there is a large number of variants of these optimization techniques published in the literature, introducing various mechanisms and methods for the improvement of the optimization procedure performance. In this study, a variant of GA with linear ranking, stochastic universal sampling, uniform mutation and uniform crossover was applied, with all parameter values based on the recommendations in the literature [31].

### 3.2 Particle swarm optimization

Particle Swarm Optimization (PSO) algorithm is a swarm-based optimization technique, inspired by the social behaviour of animals moving in large groups (particularly birds) [32]. It uses a set of particles called swarm to investigate the search space. Each particle is described by its position ( $x$ ) and velocity ( $v$ ). The position of each particle is a potential solution, and the best position that each particle achieved during the entire optimization process is memorized ( $p$ ). The swarm as a whole memorizes the best position ever achieved by any of its particles ( $g$ ). The position and the velocity of each particle in the  $k$ -th iteration are updated as

$$\begin{aligned} v[k+1] &= w \cdot v[k] + cp \cdot rp[k] \cdot (p[k] - x[k]) + \\ &\quad + cg \cdot rg[k] \cdot (g[k] - x[k]) \\ x[k+1] &= x[k] + v[k+1]. \end{aligned} \quad (3)$$

Acceleration factors  $cp$  and  $cg$  control the relative impact of the personal (local) and common (global) knowledge on the movement of each particle. Inertia factor  $w$ , which was introduced for the first time in [33] keeps the swarm together and prevents it from diversifying excessively and therefore diminishing PSO into a pure random search. Random numbers  $rp$  and  $rg$  are mutually independent and uniformly distributed in the range  $[0, 1]$ .

There are many modifications of PSO algorithm presented in literature. The early concept of PSO algorithm uses constant parameter set ( $cp, cg, w$ ), while some more recent modifications introduce variable algorithm parameter set in order to improve the overall performance of the algorithm [17, 34, 35]. In this paper, the PSO variant with all variable parameters will be used. Inertia factor  $w$  is decreased from 0.95 to 0.4, enabling thorough space search. Cognitive factor  $cp$  is also decreased during the search, from 2.5 to 0.5, enabling better exploration, while social factor  $cg$  is increased from 0.5 to 2.5, enabling better exploitation of each individual’s “knowledge” [36].

### 3.3 Artificial bee colony optimization

Artificial bee colony (ABC) algorithm, also known as Bee Colony Optimization (BCO) is another, relatively novel swarm-based numerical optimization algorithm, based on the simulation of the foraging behaviour of honey bee swarm [15, 37]. In this algorithm, the position of food source represents a possible solution of the optimization problem and the nectar amount of the food source corresponds to the value of the optimization criterion in that solution. The colony consists of three groups of bees: employed bees, onlookers, and scouts. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. In the first step, the ABC generates a randomly distributed initial population  $P(G = 0)$  of  $SN$  solutions (food source positions), where  $SN$  denotes the size of population. Each solution  $x_i (i = 1, 2, \dots, SN)$  is a  $D$ -dimensional vector, with  $D$  being the number of variables in optimization criterion. After initialization, the population of the positions (solutions) is subjected to repeated cycles,  $C = 1, 2, \dots, Cmax$ , of the search processes. An employed or onlooker bee probabilistically produces a modification on the position (solution) in her memory for finding a new food source and tests the nectar amount of the new source. While in case of real bees, the production of new food sources is based on the comparison process of food sources in a region depending on the information gathered visually by the bee, in ABC model, the artificial bees do not use any information in comparison. They randomly select a food source position and produce a modification on the one existing in their memory, using the expression:

$$v_{ij} = x_{ij} + \phi_{ij} (x_{ij} - x_{kj}) \tag{4}$$

where  $k \in \{1, 2, \dots, BN\}$  and  $j \in \{1, 2, \dots, D\}$  are randomly chosen indexes,  $BN$  is the number of employed bees and  $\phi$  is a random number in range  $[-1, 1]$ .

Provided that the nectar amount of the new source is higher than that of the previous one the bee memorizes the new position and forgets the old one. Otherwise, she keeps the position of the previous one. After all employed bees complete the search process they share the nectar information of the food sources and their position information with the onlooker bees on the dance area. An onlooker bee evaluates the nectar information taken from all the employed bees and chooses a food source with the probability related to its nectar amount. As in case of the employed bee, she produces a modification on the position in her memory and checks the nectar amount of the candidate source. Providing that its nectar amount is higher than that of the previous one, the bee memorizes the new position and forgets the old one.

An onlooker bee chooses a food source depending on the probability value associated with that food source,  $p_i$ , calculated by the following expression:

$$p_i = \frac{f(x_i)}{\sum_{n=1}^{SN} f(x_n)} \tag{5}$$

where  $f(x_i)$  is the value of the optimization criterion for solution  $i$  evaluated by its employed bee, and  $SN$  is the number of food sources which is equal to the number of employed bees ( $BN$ ). In this way, the employed bees exchange their information with the onlookers.

Employed bees whose solutions cannot be improved through a predetermined number of trials, called limit, become scouts and their solutions are abandoned. Then, the scouts start to search for new solutions, randomly. Hence, those sources which are initially poor or have been made poor by exploitation are abandoned.

The described search process is conducted until a termination criterion is satisfied; for example a maximum cycle number or a maximum CPU time.

## 4. RESULTS AND DISCUSSION

The population for every algorithm consisted of totally 30 individuals (particles, bees). The optimization process was conducted in 15 iterations (generations). The number of iterations was adopted based on the empirical analysis, which showed that after 15 iterations almost no improvement could be noticed for all three algorithms, implying that all methods converged at least to the vicinity of the optimal solution. Each individual (particle, bee) consists of four values (coordinates). These values are parameters that define the shape and position of the sigmoid membership functions in fuzzy sets. First two variables  $X_{LGO}$  and  $Y_{LGO}$  determine the shape and the position of membership functions for fuzzy variable  $LGO$ . The other two variables  $X_{LGC}$  and  $Y_{LGC}$  determine the shape and the position of membership functions for fuzzy variable  $LGC$ . As shown in Figures 3 and 4, input fuzzy variables  $LGO$  and  $LGC$  consist of three sigmoid functions. "Medium" sigmoid function is defined with two values  $X$  and  $Y$ , and other two sigmoid functions can be observed as inverse functions.

The weight coefficients  $A$  and  $B$  (see Equation 2) emphasize the significance of each component in the optimality criterion (number of empty lockages and average waiting time per ship). These coefficients can be adjusted in accordance to the desired economic performance of the overall system, defined by the management. Based on such defined criterion, fuzzy expert system can be adjusted to provide the desired system behaviour and to enable performance control, which was the main objective of this paper. If management of a lock were able to assess the economic costs of an empty lockage and costs of delays (waiting times) then coefficients  $A$  and  $B$  could be set according to the assessment. The coefficient ratios in this paper

Table 3 - FES parameter values obtained by different forms of economic criteria

FES parameter	Original FES	Optimized FES 1:1			Optimized FES 4:1			Optimized FES 1:4		
		GA	PSO	ABC	GA	PSO	ABC	GA	PSO	ABC
$X_{LGO}$	40	48.10	49.94	49.28	70.55	70.11	71.22	32.11	31.92	33.65
$Y_{LGO}$	60	69.82	69.94	69.27	91.31	91.03	90.24	59.91	60.23	61.88
$X_{LGC}$	20	17.21	16.00	18.23	8.21	8.10	9.21	69.86	68.54	66.28
$Y_{LGC}$	40	46.62	48.32	47.43	25.04	24.00	25.32	89.72	88.31	89.73

Table 4 - Comparative presentation of simulation results for different evaluation models and economic criteria

		Evaluation model					
		MWT	MNL	Original FES	Optimized FES		
					GA	PSO	ABC
Criterion 1:4	NoEL	1,410	50	746	824	822	802
	AWT [min]	4.18	3,090.85	137.3	114.17	114.56	115.94
	Criterion value	713.36	6,206.7	647.6	640.33	640.33	632.89
Criterion 1:1	NoEL	1,410	50	746	726	720	726
	AWT [min]	4.18	3,090.85	137.3	137.9	140.53	137.79
	Criterion value	1,414.18	3,140.85	883.3	863.91	860.53	863.79
Criterion 4:1	NoEL	1,410	50	746	670	670	676
	AWT [min]	4.18	3,090.85	137.3	144.19	144.19	144.44
	Criterion value	2,822.09	1,645.42	1,560.65	1,412.09	1,412.09	1,424.22

were chosen to give different significance to both parts of the economic criterion. Three variants of the economic criterion were considered, with different values of coefficients  $A$  and  $B$ . In the first variant, both coefficients were equal to one, giving equal significance to both number of empty lockages and average waiting time per ship. The second criterion variant had values of  $A = 2$  and  $B = 0.5$ , favouring the number of empty lockages in 4:1 ratio, and the third one of  $A = 0.5$  and  $B = 2$ , favouring the average waiting time per ship, in 1:4 ratio.

The values of optimal FES parameters obtained by using all three optimization algorithms are shown in Table 3. One can notice that all three algorithms converged to similar parameter values for each criterion variant. Also, it should be emphasized that the shape of the membership function varies significantly for different criterion variant, implying that fuzzy parameters depend to a large degree on desired system behaviour.

Table 4 shows the simulation results obtained using the criteria of MWT and MNL, the original fuzzy system and new, optimized fuzzy systems, for all three optimization algorithms. The number of empty lockages (NoEL), average waiting time (AWT) per ship and economic criterion value are shown for every model and every criterion variant. The results are compared by the obtained value of economic criterion defined by (2), for different variants of weight coefficients  $A$  and  $B$ . The best results for each criterion variant are typed in bold.

One can notice that the worst values for all economic criteria variants are obtained using MWT and MNL, since these models optimize only one of two parameters, i.e. average waiting time per ship and number of empty lockages, respectively. Significant improvement in criteria values is obtained using original, non-optimized FES which considers both parameters. However, it is easily observed that the best criteria values in all variants are provided by optimized FESs. In the criterion variant favouring the average waiting time (1:4), ABC showed the best performance, while GA and PSO followed, obtaining the same criterion value. In the case of equal significance of the number of empty lockages and average waiting time (variant 1:1), PSO provided the best criterion value, while the values obtained using GA and ABC were almost the same. In criterion variant 4:1, which favours the number of empty lockages, GA and PSO converged to the same value, while ABC obtained a slightly worse result. Thus, it is not possible to distinct the best optimization algorithm for universal application. We can only conclude that all three algorithms provide the best solution in some criterion variant and thus all of them can be successfully applied in such problems.

## 5. CONCLUSION

This paper presents a method for creating a fuzzy expert system that can be used as support in decision-making or in training the ship lock operators. The pa-

parameters of such a system were optimized using three popular global optimization procedures in order to minimize three different variants of the economic optimization criterion. The presented results show that all algorithms, with slight variations in criterion results, provide performance improvement, i.e. the number of empty lockages and average waiting time decrease, compared to originally proposed fuzzy expert system. Thus, we can conclude that all these algorithms can be successfully applied in this kind of transportation planning and control problems.

Fuzzy expert system, which was the object of the optimization, is one of the components of decision support system used in ship lock control. It also implements the experience of the operator interpreted by the membership functions and fuzzy rules. Once optimized according to the optimality criterion which reflects the desired economic performance, it provides suggestions for the operator how to control the operation of the ship lock. In the more automated variant of the control system, fuzzy expert system can also directly control the ship lock operation, eliminating the need for human operator. This way one can improve the overall performance of the system and decrease the probability of errors caused by the human factor.

Further research can proceed in the direction of greater complexity in lock functionality. Instead of the single-channel lock, a multi-channel lock can be considered. Such systems would need to have fuzzy input variables and fuzzy rules that are more complex. Moreover, military, service, commercial and private vessels in actual systems do not have the same priority; some classes of vessels have higher priority for using the lock than others. This could introduce additional fuzzy rules. A well-designed fuzzy expert system could serve as a valuable aid in the choice of the control action when there are more requests for lockage by a number of vessels with different priorities.

## ACKNOWLEDGMENT

This research was supported by the Ministry of Education and Science (Government of the Republic of Serbia) under Grant Number TR-36007.

Dr **ŽELJKO KANOVIĆ**

E-mail: kanovic@uns.ac.rs

**VLADIMIR BUGARSKI**

Departman za računarstvo i automatiku

Dr **TODOR BAČKALIĆ**

E-mail: tosa@uns.ac.rs

Departman za saobraćaj

Fakultet tehničkih nauka, Univerzitet u Novom Sadu

Trg Dositeja Obradovića 6, 21000 Novi Sad, Srbija

## REZIME

### OPTIMIZACIJA SISTEMA ZA UPRAVLJANJE BRODSKOM PREVODNICOM UPOTREBOM GA, PSO I ABC: UPOREDNI PREGLED

*U ovom radu predstavljeno je poređenje nekih poznatih globalnih optimizacionih postupaka pri optimizaciji ekspertskog sistema za upravljanje broskom prevodnicom. Svrha poređenja je pronalaženje najboljeg algoritma za optimizaciju parametara funkcija pripadnosti ekspertskog upravljačkog sistema baziranog na rasplinutoj (fuzzy) logici. Optimizacija je sprovedena da bi se poboljšale performanse rada prevodnice, vreme čekanja brodova i broj prevođenja na prazno. Upoređena su tri postupka: optimizacija rojem čestica, optimizacija veštačkom kolonijom pčela i genetski algoritam. Prikazani rezultati potvrđuju efikasnost sva tri postupka i poboljšanje performansi rada prevodnice, što govori u prilog njihovoj primeni u sličnim problemima transporta.*

## KLJUČNE REČI

*brodska prevodnica; ekspertski sistem sa rasplinutom logikom; optimizacija rojem čestica; optimizacija veštačkom kolonijom pčela; genetski algoritam.*

## REFERENCES

- [1] **Partenscky, H.W.:** *Inland waterways: lock installations.* (Binnenverkehrswasserbau: Schleusenanlagen - in original). Berlin: Springer; 1986.
- [2] **Bačkalić, T.:** *Traffic control on artificial waterways of limited dimensions in function of its throughput capacity (Upravljanje saobraćajem na veštačkim plovim putevima ograničenih dimenzija u funkciji njihove propusne sposobnosti - in original).* Novi Sad: University of Novi Sad, Faculty of technical sciences, PhD thesis; 2000.
- [3] **Smith, L.D., Sweeney, I.I.D.C., Campbell, J.F.:** *Simulation of alternative approaches to relieving congestion at locks in a river transportation system.* Journal of the Operational Research Society. 2009; 60:519-533
- [4] **Radmilović, Z., Maraš, V., Jovanović, S.:** *Ship lock as general queuing system with batch arrivals and batch service.* PROMET - Traffic&Transportation. 2012;19(6):343-352
- [5] **Bugarski, V., Bačkalić, T., Kuzmanov, U.:** *Fuzzy decision support system for ship lock control.* Expert Systems with Applications. 2013; 40(10):3953-3960 <http://dx.doi.org/10.1016/j.eswa.2012.12.101>
- [6] **Campbell, J.F., Smith, L.D., Sweeney, I.I.D.C., Mundy, R., Nauss, R.M.:** *Decision tools for reducing congestion at locks on the upper Mississippi river.* Proceedings of the 40th Hawaii International Conference on System Sciences. 2007; 55-58
- [7] **Kecman, V.:** *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models.* Boston, MA: Massachusetts Institute of Technology; 2001
- [8] **Kosko, B.:** *Fuzzy thinking: the new science of fuzzy logic.* New York: Hyperion; 1993



- [9] Comes, T., Hiete, M., Wijngaards, N., Schultmann, F.: *Decision maps: a framework for multi-criteria decision support under severe uncertainty*. Decision Support Systems. 2011; 52(1):108-118
- [10] Onieva, E., Milanés, V., Villagra, J., Perez, J., Godoy, J.: *Genetic optimization of a vehicle fuzzy decision system for intersections*. Expert Systems with Applications. 2012; 39(18):13148-13157. <http://dx.doi.org/10.1016/j.eswa.2012.05.087>
- [11] Teodorović, D., Vukadinović, K.: *Traffic control and transport planning: a fuzzy sets and neural networks approach*. Norwel, MA: Kluwer Academia Publishers; 1998
- [12] Castanho, M.J.P., Hernandes, F., De Re, A.M., Rautenberg, S., Billis, A.: *Fuzzy expert system for predicting pathological stage of prostate cancer*. Expert Systems with Applications. 2013; 40(2):466-470. <http://dx.doi.org/10.1016/j.eswa.2012.07.046>
- [13] Dasgupta, D., Michalewicz, Z.: *Evolutionary algorithms in engineering applications*. Berlin: Springer Verlag; 1997
- [14] Yunusoglu, M.G., Selim, H.: *A fuzzy rule based expert system for stock evaluation and portfolio construction: an application to Istanbul Stock Exchange*. Expert Systems with Applications. 2013; 40(3):908-920. <http://dx.doi.org/10.1016/j.eswa.2012.05.047>
- [15] Teodorović, D., Dell'Orco, M.: *Bee colony optimization – a cooperative learning approach to complex transportation problems*. Advanced OR and AI Methods in Transportation. Proceedings of 16<sup>th</sup> Mini-EURO Conference and 10th Meeting of EWGT. 2005 Sept.; 51-60.
- [16] He, Q., Wang, L.: *An effective co-evolutionary particle swarm optimization for constrained engineering design problems*. Engineering Applications of Artificial Intelligence. 2007; 20:89-99.
- [17] Kanović, Ž., Rapaić, M., Jeličić, Z.: *Generalized particle swarm optimization algorithm: theoretical and empirical analysis with application in fault detection*. Applied Mathematics and Computation. 2011; 217:10175-10186. <http://www.sciencedirect.com/science/article/pii/S0096300311006680>
- [18] Ting, C.J., Schonfield, P.: *Control alternatives at a waterway lock*. Journal of Waterway, Port, Coastal and Ocean Engineering. 2001; 127(2):89-96. [http://dx.doi.org/10.1061/\(ASCE\)0733-950X\(2001\)127:2\(89\)](http://dx.doi.org/10.1061/(ASCE)0733-950X(2001)127:2(89))
- [19] International Navigation Association. Inland Navigation Commission. *Guidelines and recommendations for river information services*. International Navigation Association; 2004.
- [20] Willems, C., Schmorak, N.: *River Information Services on the way to maturity*. Proc. on 32<sup>nd</sup> PIANC International Navigation Congress, Liverpool, United Kingdom, 10-14 May 2010. 2010; 1:285-297
- [21] Yager, R.R., Filev, D.P.: *Essentials of fuzzy modeling and control*. New York: John Wiley and Sons; 1994
- [22] Zimmermann, H.J.: *Fuzzy set theory and its applications*. 4<sup>th</sup> ed. Dordrecht: Kluwer Academic Publishers; 2001
- [23] Camps-Valls, G., Martín-Guerrero, J.D., Rojo-Alvarez, J.L., Soria-Olivas, E.: *Fuzzy sigmoid kernel for support vector classifiers*. Neurocomputing. 2004; 62:501-506
- [24] Jang, J-S.R., Sun, C-T., Mizutani, E.: *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. New Jersey: Prentice Hall; 1997
- [25] Nguyen, H.T., Sugeno, M.: *Fuzzy systems: modelling and control*. Dordrecht: Kluwer Academic Publishers; 1998
- [26] Jantzen, J.: *Foundations of fuzzy control*. New Jersey: John Wiley and Sons; 2007
- [27] Lancaster, S.: *Fuzzy logic controllers*. Portland: Mas-eeh College of Engineering and Computer Science at PSU; 2008
- [28] Collette, Y., Siarry, P.: *Multiobjective optimization: principles and case studies*. Berlin: Springer; 2004
- [29] Rao, R.V., Patel, V.: *Multi-objective optimization of two stage thermoelectric cooler using a modified teaching-learning-based optimization algorithm*. Engineering Applications of Artificial Intelligence. 2013; 26(1):430-445. <http://dx.doi.org/10.1016/j.engappai.2012.02.016>
- [30] Holland, J.: *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press; 1975
- [31] Michalewicz, Z.: *Genetic algorithms + data structures = evolution programming*. 3<sup>rd</sup> ed. Berlin: Springer Verlag; 1999
- [32] Kennedy, J., Eberhart, R.C.: *Particle swarm optimization*. Proceedings of IEEE International Conference on Neural Networks, Perth, Australia. 1995; 1942-1948
- [33] Shi, Y., Eberhart, R.C.: *Empirical study of particle swarm optimization*. Proceedings of IEEE International Congress on Evolutionary Computation. 1999; 3:101-106
- [34] Clerc, M., Kennedy, J.: *The particle swarm: explosion, stability and convergence in a multidimensional complex space*. IEEE Transactions on Evolutionary Computation. 2002; 6(1):58-73
- [35] Rapaić, M., Kanović, Ž.: *Time-varying PSO: convergence analysis, convergence related parameterization and new parameter adjustment schemes*. Information Processing Letters. 2009; 109(1):548-552 <http://www.sciencedirect.com/science/article/pii/S0020019009000350#>
- [36] Ratnaweera, A., Saman, K.H., Watson, H.C.: *Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients*. IEEE Transactions on Evolutionary Computation. 2004; 8(3):240-255
- [37] Karaboga, D.: *An idea based on honey bee swarm for numerical optimization*. Technical Report- TR06. Kayseri, Turkey: Erciyes University; 2005

