⊗ANS

Fuzzy Logics as an Integral Part of Evolutionary Algorithms

Amit Thakur* and Umasankari Kannan

Bhabha Atomic Research Centre, Reactor Physics Design Division, Mumbai 400085, India

Abstract — *Evolutionary algorithms play an important role for solving various optimization problems related to fuel management in reactor physics like core loading pattern optimization (LPO) or refueling. In general, all algorithms make a sample of solution candidates and evaluate the fitness of all candidates, and then the candidates with better fitness value are used to generate the next sample of solution candidates. In optimization algorithms, internal parameters [like population size, weighting factor in estimation of distribution algorithm (EDA) and population size, cross-over rate, etc., in Genetic Algorithm (GA)] have a stiffness problem as the value of these parameters is fixed at the first generation and is not being changed subsequently. However, the flexibility of changing the value of even one internal parameter during the generations will make the algorithm more efficient. In this paper we propose that fuzzy logics can be used in an innovative way to eliminate the stiffness problem related to internal parameters in evolutionary algorithms. As a test case, EDA for initial core LPO of the advanced heavy water reactor is chosen, and the use of fuzzy logics has shown a significant improvement in the algorithm's performance. The appropriate value of weighting factor α in EDA has been predicted using fuzzy logics in each generation, and this has resulted in efficiency improvement of the algorithm. The improved methodology is expected to give better performance with other optimization algorithms, such as the GA or Ant Colony Optimization Algorithm, etc.*

Keywords — *Loading pattern optimization, advanced heavy water reactor, estimation of distribution algorithm, fuzzy logics.*

Note — *Some figures may be in color only in the electronic version.*

I. INTRODUCTION

Fuel management in nuclear reactors is an involved mathematical problem where contradicting parameters need to be optimized simultaneously. At the initial stage of any reactor or during refueling, one has to load different types of fuel clusters in a given number of core locations. The search-space of the loading pattern optimization (LPO) problem related to nuclear reactor is very vast due to the large number of lattice locations and fuel types. In these combinatorial optimization problems, the main aim is to obtain the loading pattern (LP) that has the best multiplication factor (*k*-effective) and good safety margins. The gradient-based methods[1] have limited success to comprehensively scan the search-

space. The search-space has multiple maxima/minima, and therefore, the search for a global optimum solution is difficult using conventional methods.[1–3] Evolutionary algorithms like Genetic Algorithm[4,5] (GA), Particle Swarm Optimization,[6] Estimation of Distribution Algorithm[7,8] (EDA), Gravitational Search Algorithm[9] (GSA), and Ant Colony Algorithm[10] have been successfully applied for core reloading optimization problems of water-cooled reactors.

The internal parameters play an important role in these evolutionary algorithms. The population size *N* is the most common internal parameter for these algorithms, and it has a fixed value for all generations. Similarly, other internal parameters are also fixed in the beginning for all the algorithms. Keeping a fixed value of any internal parameter for all generations is termed a stiffness problem. In our earlier work[11] we have used EDA and GA for initial core

*E-mail: thakur@barc.gov.in

LPO of the advanced heavy water reactor[12,13] (AHWR). It was observed that with a proper choice of internal parameter population size N and weighting factor α in EDA resulted in better optimized LP.

In this paper, we have used fuzzy logics to define a new value of internal parameters during each generation, thereby removing this stiffness problem. We have observed that including fuzzy logics as an integral part of the EDA algorithm becomes extremely efficient. This paper is arranged in the following manner. Section II describes the working principle of evolutionary algorithms and the role of internal parameters. Section III describes our earlier[11] LPO studies using EDA for the initial core of the AHWR. The development of fuzzy logic controllers (FLCs) and how they are different from previous FLCs is described in Sec. IV. The results and comparison of fuzzy logics–based EDA with the basic form of EDA are discussed in Sec. V. Section VI discusses the conclusions.

II. PRINCIPLE OF EVOLUTIONARY ALGORITHMS

Evolutionary algorithms start with the generation of population N of candidate solutions using specific sample generation criteria. These N candidates are then evaluated for their fitness value. The sample generation criteria are then updated for next generation by considering the fitness of few better solutions. Figure 1 describes the general working of evolutionary algorithms. In this paper, we have considered the use of fizzy logics in EDA; therefore, for completeness the working of EDA and the role of internal parameters in EDA are described below.

In case of EDA, the distribution function (DF) describes the allocation of various fuel types in core locations. In each generation, DF is used to generate

Fig. 1. Working of any evolutionary optimization algorithm.

a sample of N (population size) candidates (LPs) of solution pool and is updated at the end of each generation. For example, consider a LPO problem with two fuel types (type 1 and type 2) in a core with 100 lattice locations, keeping $N = 50$ and a uniform DF of 0.6 for all 100 core locations at first generation. For every core location, a random number will be generated and its value is compared with DF for that core location. If it is less (than 0.6 for first generation), type 1 will be loaded, otherwise type 2 will be loaded. Therefore, a core LP generated this way will have ~60% of type 1 fuel and 40% of type 2 fuel. The fitness value of all the N candidates is then evaluated. The best M out of N LPs based on their fitness values are used to update the DF corresponding to each location j in the core using Eq. (1):

$$\mathrm{DF}_j(t+1) = \mathrm{DF}_j(t) \cdot (1-\alpha) + \alpha \cdot Y_j , \qquad (1)$$

where

$$Y_j = \frac{1}{M} \cdot \sum_{i=1}^{M} X_{ij}(t) \qquad (1a)$$

and where α is weighting factor with a value between 0 and 1. The X_{ij} for the i'th LP (in N) is 0 or 1 corresponding to loading type 2 fuel or type 1 fuel, respectively, in the j'th location of core. The Y_j for j'th location is the average of X_{ij} for best M candidates. The structure of $\mathrm{DF}_j(t)$ and Y_j is same, and t represents the generation.

In the first generation, a uniform guess DF is considered which is then updated after each generation. If convergence is achieved, the DF will achieve either a value of 0 or 1 corresponding to each core location.

The value of $\alpha \cdot Y_j$ will decide the updating of the DF. If value of α is large, then the term $\alpha \cdot Y_j$ will have more weight compared to the other term $\mathrm{DF}_j(t) \cdot (1-\alpha)$. This shows that with higher value of α, the algorithm becomes greedy as the DF now updates more in the direction of best solutions of previous generation. We can make the algorithm greedier by choosing small M. For example, if we choose only 1% of best solutions based on fitness values, with a large value of $\alpha = 0.8$, then in the next generation most of the LPs will be generated that are similar to the top 1% LPs of the previous generation. Therefore, the chances are more that we achieve local minima/maxima than global.

Therefore, the choice of internal parameters greatly affects the direction of search for global maxima/minima. For example, in our earlier work[11] we considered
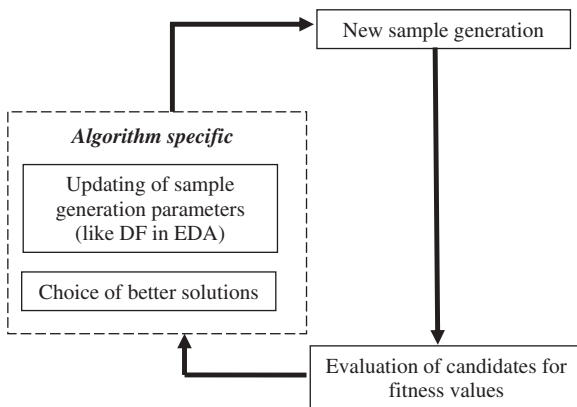
different values of α (0.05, 0.1, and 0.5) and N (24, 240, and 1200) and concluded that α = 0.5 and N = 1200 is the best suitable value for the AHWR initial core LPO problem. The internal parameters show a stiffness problem as their value is fixed in the very first generation, and numerical exercises are required to estimate the best fixed values for these internal parameters, as we have shown in our earlier work. The evolutionary algorithms do not have this facility to choose the suitable value of internal parameters during each generation. In this paper, we have integrated fuzzy logics in EDA to update the value of α in each generation for the AHWR LPO problem.

In the case of GA, there is no DF in each generation; however, the better candidates are chosen using tournament selection and are used for reproduction of new solutions using cross-over operation in the mating pool for the next generation. Population size, tournament selection, and mutation rate are different internal parameters in GA. The similar nature of evolutionary algorithms can also be seen in Fig. 1.

There have been continuous efforts to explore new optimization algorithms in search of global maximum or minimum solution. However, less stress has been given on the stiffness problem (keeping the same value of internal parameters during all generations) and the choice of internal parameters in optimization algorithms. To overcome this stiffness problem, a choice should be available for internal parameters at each generation based on the improvement of objective function (OF) values from previous generations. It will be very difficult to define a function for internal parameters that will decide its appropriate value at each generation due to the inherent fuzziness involved in the process. Therefore, the application of fuzzy logics[14] should be more suitable for this kind of problem. In Ref. 15, a fuzzy control system based on GSA (Ref. 9) is proposed to reduce the sensitivity due to parameter variation. Kim et al.[16,17] have used fuzzy rule–based system along with artificial neural networks for LPO of the pressurized water reactor (PWR). The use of fuzzy logics and heuristics will be more effective in the case of algorithms like Simulated Annealing.[18] However, Kim et al. have not discussed the actual application on evolutionary algorithms. Saeidi-Khabisi and Rashedi[19] have used fuzzy logics in GSA to control its internal parameter (gravitational constant G). They have used few standard functions to show the performance improvement of fuzzy GSA (FGSA). Aghaie and Mahmoudi[20] used the same approach for LPO of the PWR. They have compared the GSA results with FGSA. A FLC (Ref. 20) was developed

to choose the value of α from 0 to 1 in FGSA. However, in GSA the α was fixed as 0.001. The recipe for FGSA shows fast convergence rate and improvement in average values over various simulations, but overall, there is small improvement in efficiency for FGSA.

In this paper, we have used fuzzy logics in similar way[20] to choose the internal parameter α in EDA for the initial core LPO of the AHWR. However, the FLC (Refs. 20 and 21) is developed considering different aspects of the algorithm that are described in detail in Sec. IV. We have observed significant performance improvement for our problem and believe that similar results are expected for other algorithms and other optimization problems as well.

III. INITIAL CORE LPO OF THE AHWR USING EDA

In this paper we have considered the initial core LPO using EDA as a test case for the use of fuzzy logics in deciding the internal parameters at each generation. The results are compared with our earlier reported work.[11] Therefore, a short description of our earlier work using EDA is given below.

The AHWR is a pressure tube–type thermal reactor with light water as coolant and heavy water as moderator. It has full-length channels and on-power refueling. The initial core LPO of the AHWR consists of placing at least two types of clusters in 444 core locations. The two types of clusters have a fissile content of ~2%, and Gd has been used for making the clusters differentially reactive. This combinatorial optimization problem requires 2^{444} different core simulations for finding the best LP. The core symmetry has been used to reduce the problem size to 2^{62}. The two types of clusters used are named type 1 and type 2. The type 2 fuel is less reactive (due to the presence of Gd) than the type 1 fuel. The EDA as described in Sec. II has been used to solve this problem.

In present studies, we have defined OF (for fitness value) in same way as in our earlier work[11] so that the comparison can be made. The OF for the initial core LPO has been defined to maximize k-effective and keep maximum channel power (MCP), maximum mesh power (MMP), and worth of shutdown system (SDS) in the design limit. The OF is defined using the penalty method[8,11]:

$$
\begin{aligned}
\mathrm{OF} = A_1 \cdot k - \mathit{eff} - A_2 \cdot (\mathrm{MCP} - 2.6) - A_3 \\
\cdot (\mathrm{MMP} - 200) - A_4 \\
\cdot (63.0 - \mathrm{worth\ of\ SDS} - 1) ,
\end{aligned} \tag{2}
$$

where A_1, A_2, A_3, and A_4 are constants. In this optimization problem, the value of A_1 is taken as 1. For optimization studies, worth requirements of 63.0 mk have been considered for SDS-1:

1. if the MCP of a LP is <2.6, then $A_2 = 0$, else $A_2 = 0.384$

2. if the MMP < 200, then $A_3 = 0$, else $A_3 = 0.05$

3. and worth of SDS-1 > 63.0, then $A_4 = 0$, else $A_4 = 0.333$.

Hence, an LP having higher k-effective will be more preferred than an LP with higher safety margins. The N candidates are evaluated by doing three-dimensional (3-D) diffusion calculations using the in-house-developed 3-D diffusion theory code Flux Expansion Method in Nodal Analysis, and the k-effective and other operational parameters are evaluated to estimate OFs for all N candidates.

In our earlier studies,[11] multiple simulations have been carried out considering fixed different values of α and N. These values were not changed during the generations. Different values of α and N have resulted in different optimized solutions. The value of α has been considered as 0.05, 0.1, and 0.5, and the population size N has been considered as 24, 240, and 1200. Two different initial DFs were also studied. The final conclusion is that the population size of 1200 and α value of 0.5 are most suitable for the AHWR LPO problem. A conclusion that can be drawn is that the low value of α in EDA does not allow the convergence to take place. However, a high value of α makes the algorithm greedy as described in Sec. II. This problem is tackled by using a higher population size of 1200. However, using a large population size makes the algorithm computationally less efficient.

In the present paper, we have used fuzzy logics to decide the value of α at each generation for EDA. There are other internal parameters like M and N that could also be decided based on fuzzy inputs in each generation. For the sake of comparison with earlier results, we have kept M and N the same and used fuzzy logics for one parameter α only. By allowing too many parameters to be decided by fuzzy logics, the complications will also grow. If the value of one parameter is not chosen properly, unknown errors will be induced that will be difficult to be located.

IV. FUZZY LOGICS IN EDA

Different evolutionary algorithms have been tried along with fuzzy logics to control the parameters aiming for better

performance on reactor and non-reactor optimization problems.[15,21,22] Saeidi-Khabisi and Rashedi[19] have developed a FLC to decide the value of α used in GSA. This α will decide the value of gravitational constant G which affects the exploration of search-space. Therefore, this α is playing a similar role for both algorithms (EDA and GSA). Aghaie and Mahmoudi[20] used the similar FLC to solve the LPO of the PWR. The value of α has been decided based on a few performance indicators from earlier generations that have been defined as:

1. iteration number t

2. diversity div:

$$div = \frac{r_{avg} - r_{min}}{r_{max} - r_{min}} \quad , \tag{3}$$

where r_{avg}, r_{max}, and r_{min} are the average, maximum, and minimum distance between the agents and best results, respectively, in GSA. The parameter div decides the diversity in the search-space.

3. progress $prog$:

$$prog = \frac{fit_{avg}(t-1) - fit_{avg}(t)}{fit_{avg}(t)} \quad , \tag{4}$$

which defines the progress toward better solution in next generation.

4. $\alpha(t-1)$.

If div approaches 0, the population has low diversity, and if the average value of fitness has improved, then $prog$ is positive, else it is negative or zero.

In Ref. 20, to design a FLC, four performance indicators are used as inputs and $\alpha(t)$ is the desired output. However, in this paper with EDA, we have used only two performance indicators as described below:

1. *Iteration number (IT) or t*: In general, the $\alpha(t)$ value should depend on IT. In initial generations, $\alpha(t)$ should be small, and in later generations $\alpha(t)$ should increase.

2. *Improvement factor (IMP)*: IMP is similar to $prog$ defined in Ref. 20. However, IMP is chosen based on the best value of OF for two generations rather than taking the average value[20] of OF for two successive generations.

The other parameters like *div* and $\alpha(t-1)$ are not considered because $\alpha(t-1)$ does not play any significant role for deciding the value of $\alpha(t)$. In general, $\alpha(t)$ should be independent of $\alpha(t-1)$. Similarly, we did not use *div* as an indicator because the algorithm takes care of diversity on its own by searching various regions of search-space.

We have considered two different ways of defining IMP as given below:

1. *Based on successive iterations:*

$$\text{IMP}_S = \frac{\text{OF}_{best}(t) - \text{OF}_{best}(t-1)}{y} \ . \tag{5}$$

The term IMP_s has been based on improvement observed in two successive generations, and there may be a case when the OF_{best} observed in earlier generations may be higher. This may affect the progress of the algorithm. Therefore, another way was thought of to define an IMP: IMP_B.

2. *Based on best so far:*

$$\text{IMP}_B = \frac{\text{OF}_{best}(t) - \text{OF}_{best}(0 \to t-1)}{y} \ , \tag{6}$$

where the best value of present OF is compared with the maximum value of all previous generations. This way the α value will be increased only when overall improvement is observed.

We have considered a test case of the AHWR initial core LPO, where the OF is defined for the AHWR problem as given in Eq. (2) and *y* is used to normalize the factor. We have done a random simulation with a small value of α to decide the value of *y*. In this case, based on two numerical experiments, it is considered as 6.0.

In this paper, we have used both ways to define IMP and compared the results with earlier published work[11] where a single value of α is considered.

IV.A. Membership Function for FLC

The fuzzy set theory has been used, and a set of rules have been formed in the form of IF-THEN statements to define the value of $\alpha(t)$. Triangular membership functions μ have been defined for all parameters IT, IMP, and $\alpha(t)$. The linguistic variables describing IT and $\alpha(t)$ are SMALL, MEDIUM, and LARGE. For IMP, the linguistic variables are NEGATIVE, ZERO, and POSITIVE. The

triangular membership function for $\alpha(t)$, IT, and IMP are shown in Figs. 2, 3, and 4, respectively. The range of IMP is −0.65 to +0.65. We have chosen the value of $\alpha(t)$ in a range from 0.01 to 0.65. If the IMP is POSITIVE, $\alpha(t)$ should be MEDIUM or LARGE depending on IT (or *t*). The range for IT is considered between 2 and 100. The fuzzy sets of rules are defined in Table I. A few fuzzy rules are explained here:

1. *Rule 3 in Table I*: If IT is SMALL and IMP is POSITIVE, then α should be MEDIUM (not LARGE). This is because, in the initial stage, if the best value of OF has improved, and now a large value of α may result in premature convergence.

2. *Rule 5 in Table I*: If IT is MEDIUM and IMP is ZERO then α is MEDIUM. This is because, in evolutionary methods, in the initial stage the value of OF increases very fast. But after a few iterations, very
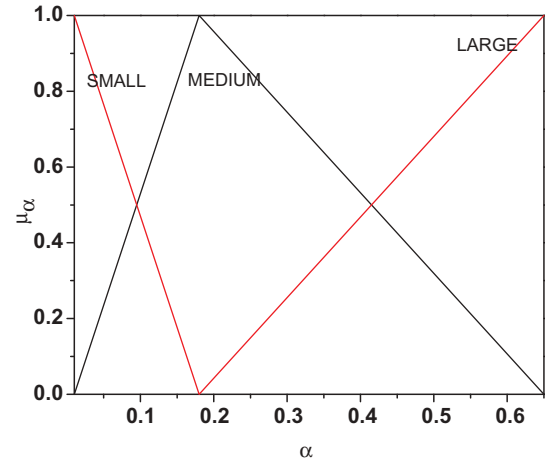

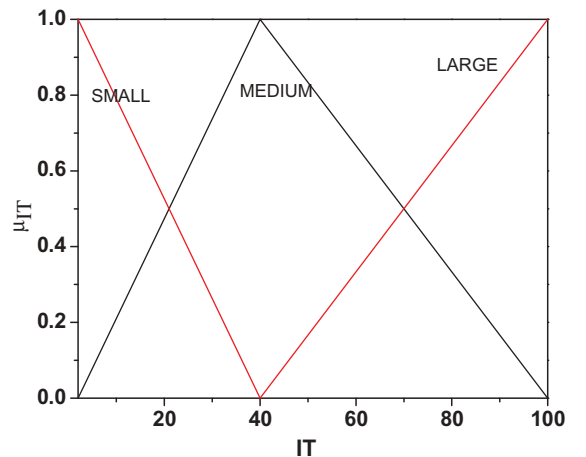Fig. 2. Membership function for α.

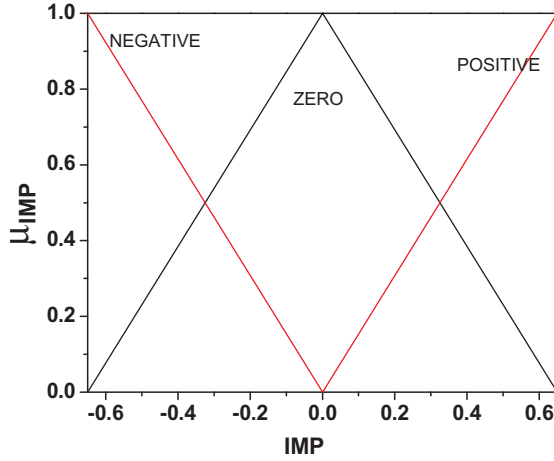
Fig. 3. Membership function for IT.

Fig. 4. Membership function for IMP.

small improvement is observed. The improvement may be small in this region but its weightage should be more. Therefore, the value of α should be MEDIUM (and not SMALL).

Similarly, other rules are also drawn.

IV.B. FLC for EDA

An integrated computer code has been developed where FLC has been made as a subroutine to EDA. The EDA is used in same way as is described in Secs. I and II and in Ref. 11. However, unlike our earlier work in Ref. 11, where α is fixed, here the value of α is calculated at each generation based on the previous generation's data. The value of α for first two generations has been considered as 0.05. The main reason for this is, in the initial two generations, IMP cannot be defined and a large value of α may lead to

premature convergence. For de-fuzzification and obtaining a unique value of α at each generation, the center of gravity (COG) approach has been implemented. A small change in the COG approach has been done to keep α large whenever IMP is positive and small when IMP is negative. It is to be noted that to reduce the probability of premature convergence, a special criterion is used. If during two consecutive generations IMP = 0 (same value of OF_{best}), then a very small value of α (0.05) is chosen. By using this technique, the algorithm searches more area of search-space. A flow diagram of the FLC-integrated EDA is given in Fig. 5.

V. RESULTS AND DISCUSSION

The integrated code (EDA with FLC) has been tested for the AHWR initial core LPO problem. It was observed from earlier results[11] of EDA (without FLC), that the best value of α for this problem is 0.5 and a population size of 1200 is required. A population size of 240 and α = 0.5 gave decent results, but with 1200 the same optimized LP as observed by GA has been generated. Further, dependence was observed on initial DF. Therefore, for a straightforward comparison, it was decided to perform the same calculations[11] considering two initial DFs with the integrated code (EDA with FLC) with a smaller population size of 240. It should be noted that both the DFs are defined such that the type 1 cluster, being more reactive, should have more numbers than the type 2 cluster. The details of the initial DF considered are given as follows:
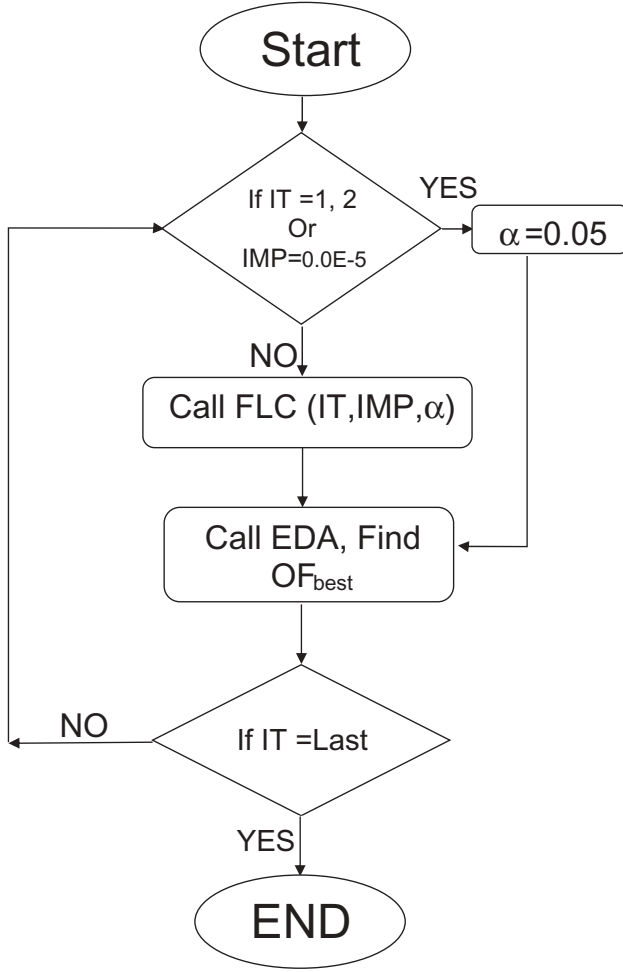
1. DF1 is such that the LP generated by it should have ~60% of type 1 fuel clusters and ~40% of type 2 fuel clusters.[11]

TABLE I

Fuzzy Rules for Defining α Using Linguistic Inputs

Rule		Iteration Number	Improvement Factor		α(*t*)
1	IF	SMALL	NEGATIVE	THEN	SMALL
2		SMALL	ZERO		MEDIUM
3		SMALL	POSITIVE		MEDIUM
4		MEDIUM	NEGATIVE		SMALL
5		MEDIUM	ZERO		MEDIUM
6		MEDIUM	POSITIVE		LARGE
7		LARGE	NEGATIVE		SMALL
8		LARGE	ZERO		MEDIUM
9		LARGE	POSITIVE		LARGE

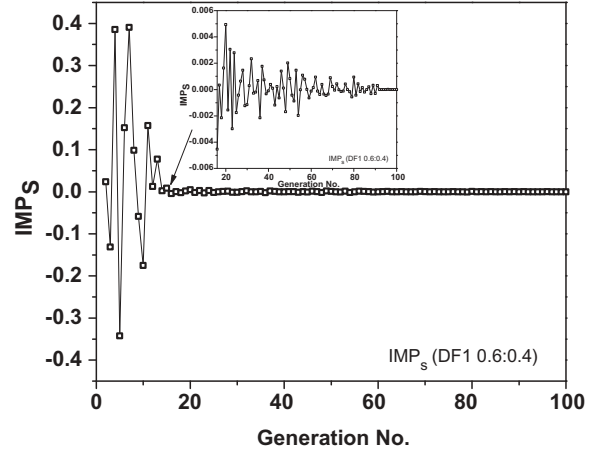Fig. 5. FLC integrated with EDA.

Fig. 6. Value of IMP_{sB} with each generation using FLC (IMP_s approach) with DF1 (0.6:0.4). The smaller variations after 18 generations are given in the inset.

However, the variations reduce significantly after 20 generations (−0.004 to +0.003). This shows that after few generations, the algorithm is nearing convergence. After ~85 generations, IMP is 0 most of time and convergence is achieved. Figure 7 shows the variation of α with generations. The value of α is generated in two regions mainly below 0.18 and above 0.18. This is based on negative IMP and positive IMP, respectively. As only two generations are considered, a zig-zag kind of curve is seen. It is also to be noted that after few iterations (~20) even with a very small positive, a negative value of IMP gives a large variation in α. This is desired, as in the later generations even a small IMP should give a large α and vice versa. The best LP achieved is the same as achieved with our earlier study[11] using population size $N = 1200$ and $α = 0.5$.

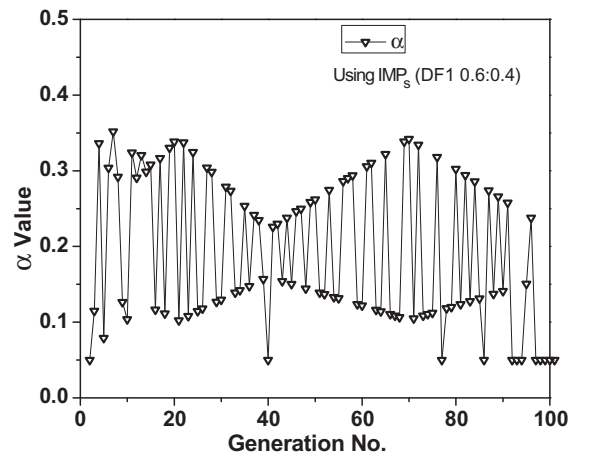2. DF2 is such that the LP generated by it should have ~80% of type 1 fuel clusters and ~20% of type 2 fuel clusters.[11]

As described in Sec. III, two different ways to define the IMP function—IMP_s (approach 1) and IMP_B (approach 2)—have been considered. Therefore, the results of both the approaches have been presented.

V.A. Approach 1 Using IMP_S

In this approach, IMP is defined based on improvement in two successive generations. Therefore, even if OF_{best} is having a higher value in initial generations, the variations are seen in two successive generations only to estimate α. The results obtained for the two initial DFs are appended below:

1. *Using DF1 (0.6:0.4)*: Figure 6 shows the variation of IMP_S with generations. It is observed that the initial variations in IMP are quite large (−0.35 to +0.4).

Fig. 7. Value of α with each generation using FLC (IMP_s approach) with DF1 (0.6:0.4).

⊗ANS

2. *Using DF2 (0.8:0.2)*: It was observed from Ref. 11 that the best optimized solution has 356 type 1 clusters and 88 type 2 clusters. This is very near to the initial DF (0.8:0.2) as compared to DF1 (0.6:0.4). Therefore, lesser variations are expected here for IMP and α. Figures 8 and 9 show the variation of IMP and α with generations. Comparing Fig. 8 with Fig. 6 and Fig. 9 with Fig. 7, it is evident that the convergence is achieved much earlier (~60 generations) in this case, as expected. Further, the variations are also significantly lower. However, the best LP achieved is the same as observed with DF1.

V.B. Approach 2 Using IMP$_B$

In this approach, IMP is defined based on the OF$_{best}$ achieved in generations so far. Therefore, for most of the generations, IMP will be negative. A positive value of IMP will indicate that a higher OF is achieved, and because of nature of IMP, smaller variations are observed in IMP. The results obtained for two initial DFs are appended below:

1. *Using DF1 (0.6:0.4)*: Figure 10 shows the variation of IMP$_B$ with generations. It is observed that the initial variations in IMP are quite large (−0.2 to +0.4). However, the variations reduce significantly after 18 generations (−0.005 to +0.003). This shows that after few generations, the algorithm is nearing convergence. After ~85 generations, IMP is 0 most of time and convergence is achieved. Figure 11 shows the variation of α with generations. The value of α is mostly in the lower region (<0.18). This is because mostly negative IMP is observed, and here the variation in α also is not

Fig. 9. Value of α with each generation using FLC (IMP$_s$ approach) with DF2 (0.8:0.2).

Fig. 10. Value of IMP$_s$ with each generation using FLC (IMP$_B$ approach) with DF1 (0.6:0.4). The smaller variations after 18 generations are given in the inset.

so zig-zag in nature. Similar to earlier studies in this section, a very small positive or negative value of IMP gives a large variation in α. The best LP achieved is the same as achieved with earlier reported work[11] using population size $N = 1200$ and $\alpha = 0.5$.

2. *Using DF2 (0.8:02)*: Similar to approach 1, lesser variations are expected here for IMP and α because of DF2. Figures 12 and 13 shows the variation of IMP and α with generations. Comparing Fig. 12 with Fig. 10 and Fig. 13 with Fig. 11, it is evident that the convergence is achieved much earlier in this case, as expected. Further, the variations are also significantly lower. However, the best LP achieved is same as observed with DF1. It is also observed that approach 2 required more numbers of generations than approach 1 to achieve convergence.
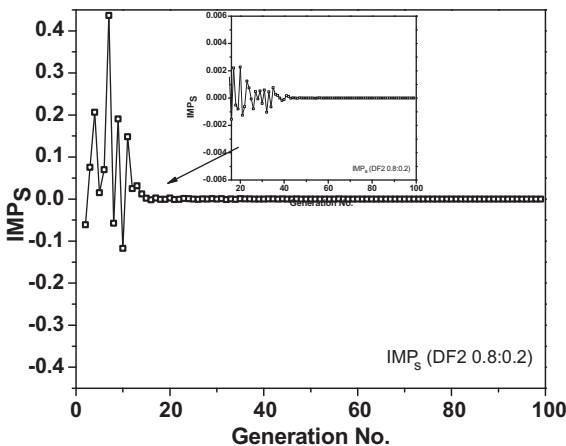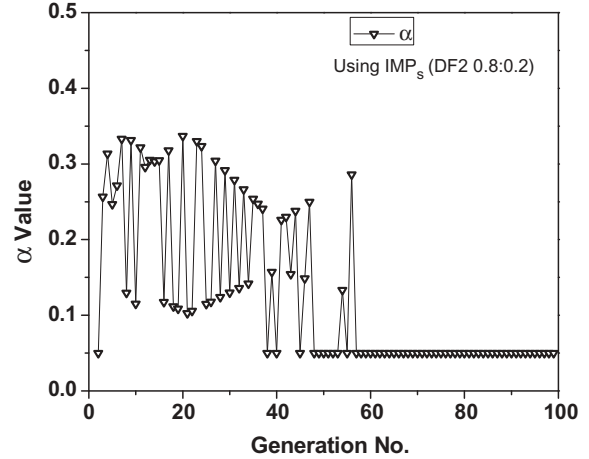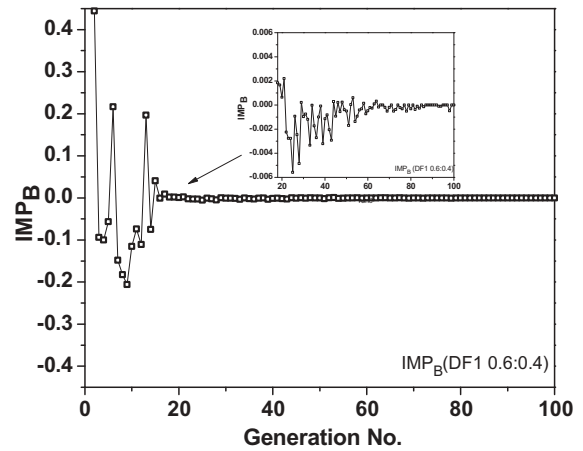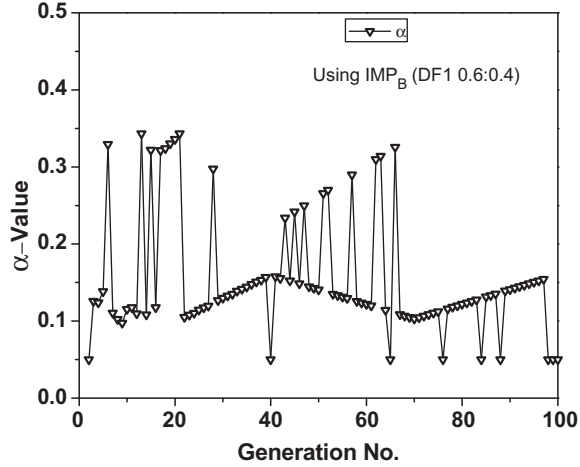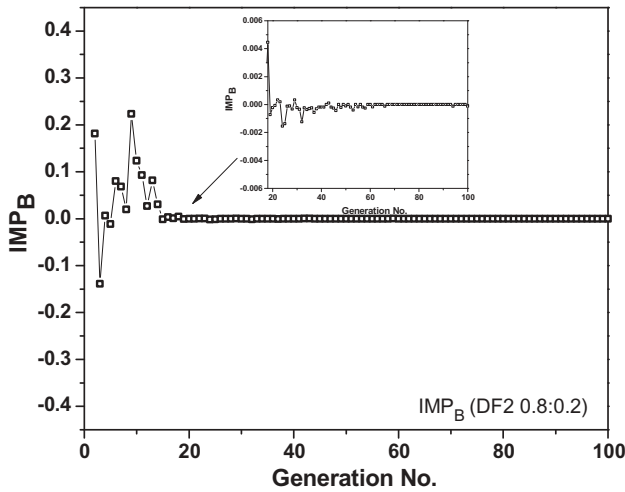

Fig. 8. Value of IMP$_s$ with each generation using FLC (IMP$_s$ approach) with DF2 (0.8:0.2). The smaller variations after 18 generations are given in the inset.

Fig. 11. Value of α with each generation using FLC (IMP_B approach) with DF1 (0.6:0.4).

Fig. 12. Value of IMP_{sB} with each generation using FLC (IMP_B approach) with DF2 (0.8:0.2). The smaller variations after 18 generations are given in the inset.
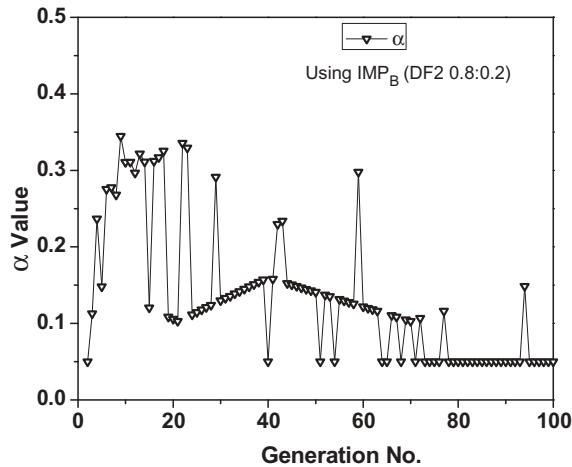

Fig. 13. Value of α with each generation using FLC (IMP_B approach) with DF2 (0.8:0.2).

It is clear from the above results that a significant improvement in the algorithm is achieved after using FLC (using both approach 1 and approach 2) as now we have reached the same LP with 240 population size N, which was achieved in 1200 population size as described in Ref. 11. And now we are able to generate same optimized LP with lesser computations.

Figure 14 compares the best value of OF (for $N = 240$) with each generation for DF1 (0.6:0.4) for the two cases (approach 1 and approach 2) with the earlier results of EDA (best α = 0.5) without any FLC. It is observed that the fastest convergence is achieved in EDA without FLC because of the stiffness problem in α. The fastest convergence may result in trapping into the local minima. It is also to be noted that EDA with FLC for both approach 1 and approach 2 has converged to a higher value of OF. Similarly, Fig. 15 compares the results for DF2 (0.8:0.2). Here it is observed that fewer numbers of generations are required for all cases for achieving convergence as compared to DF1 (0.6:0.4). Similar to Fig. 14, the fastest convergence is achieved for the case of EDA without FLC. However, the converged value is the same for all the cases. It is to be noted from the comparison of approach 1 and approach 2 that more numbers of generations are required for approach 2 than for approach 1. However, based on this, it is not possible to conclude that approach 1 is better than approach 2. Therefore, another exercise was done with a lower population size of 120 for both the cases. Figure 16 shows OF behavior with both approach 1 and approach 2 with DF1. From Fig. 16 it is observed that the OF has converged to maximum value of 1.009479 in the case of approach 1 (IMP_s).
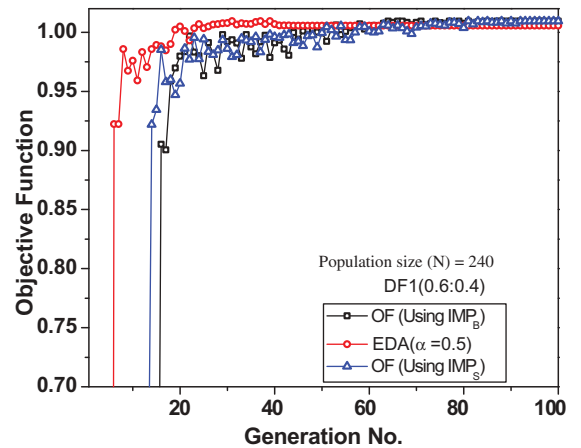

Fig. 14. Value of OF (best) with each generation with DF1 (0.6:0.4) for population size $N = 240$.
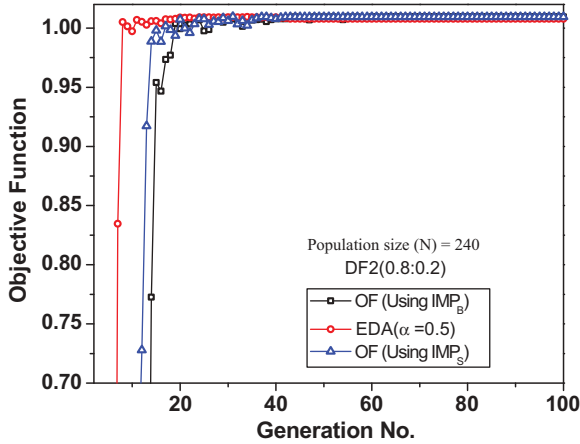
Fig. 15. Value of OF (best) with each generation with DF2 (0.8:0.2) for population size $N = 240$.
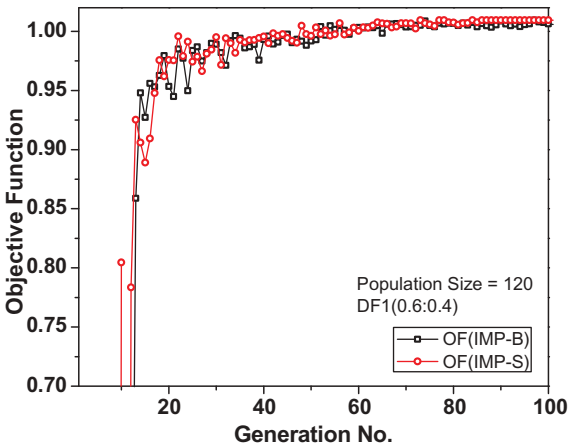

Fig. 16. Value of OF (best) with each generation for population size of 120 and with DF1 (0.6:0.4).

TABLE II

Comparison of EDA Using FLC and Simple EDA (Without FLC)

Algorithm	Population/ Distribution Function	Objective Function (Best)
EDA ($\alpha = 0.05$) EDA ($\alpha = 0.1$) EDA ($\alpha = 0.5$)	240/DF1 (0.6:0.4)	0.99536 1.00740 1.00953
EDA ($\alpha = 0.05$) EDA ($\alpha = 0.1$) EDA ($\alpha = 0.5$)	240/DF2 (0.8:0.2)	1.00720 1.00953 1.00948
EDA with FLC Approach 1 (IMP_S)	240/DF1 (0.6:0.4) 240/DF2 (0.8:0.2) 120/DF1 (0.6:0.4) 120/DF2 (0.8:0.2)	1.00953 1.00953 1.00948 1.00953
Approach 2 (IMP_B)	240/DF1 (0.6:0.4) 240/DF2 (0.8:0.2) 120/DF1 (0.6:0.4) 120/DF2 (0.8:0.2)	1.00953 1.00953 1.00947 1.00953

The convergence has not been achieved in approach 2 (IMP_B). However, the best value in 100 iterations is 1.009478. Further, the best value in approach 1 is very good and comparable to earlier results with population size of 240. Therefore, for the present, case approach 1 seems more efficient. The same exercise also was carried out for DF2 (0.8:0.2), and the same best OF (1.00953) to population size of 240 was observed. The comprehensive results are given in Table II. It is evident from Table II that EDA with FLC, a population size of 120 is also good and no external variation of parameters is needed.

VI. CONCLUSIONS

Optimization algorithms have various internal parameters and fixing them in the beginning creates a stiffness problem. In this paper we have tried to address this problem by specially designing a fuzzy logics–based controller. By using this FLC, the suitable value of internal parameters can be chosen at each generation. The technique of modifying α (weighting factor) using fuzzy logics during each generation has been successfully applied to the AHWR initial core LPO, and the algorithm now has shown better performance considering the same OF. In the present case, we have used FLC to choose only one parameter α in EDA. We have defined IMP, which is based on best value of OF, in two generations rather than taking the average value of OF from one generation to other generation.[20] It can be concluded that by eliminating the stiffness problem for parameter α, the algorithm can be made more efficient. By using FLC in EDA, the requirement of multiple initial simulations for choosing best value of α is also eliminated, which results in reduction of time and improvement in computational efficiency. The algorithm is now self-adaptive in nature for choosing the best value of α at each generation. Although this work deals with eliminating the stiffness problem of α in EDA, treatment of the stiffness problem of other parameters, like M and N, in EDA using fuzzy logics can also be considered. It is worth mentioning that by applying fuzzy logics to choose various internal parameters in other evolutionary algorithms, such as GA, ACO, etc., and in other LPO problems with more

types of fuel clusters may lead to better results and efficient performance.

Acknowledgments

The authors are thankful to Baltej Singh, Reactor Physics Design Division (RPDD), Bhabha Atomic Research Centre (BARC) for his guidance. The authors are also thankful to Vibhuti Duggal and Kislay Bhatt of Computer Division (CD), BARC for their help in running the code in the ANUPAM system. The authors are also thankful to the reviewers for critically reviewing the manuscript. Their suggestions have helped in the improvement of this manuscript.

References

1. K. BALAKRISHNAN and A. KAKODKAR, "Optimization of the Initial Fuel Loading of the Indian PHWR with Thorium Bundles for Achieving Full Power," *Ann. Nucl. Energy*, **21**, 1 (1994); https://doi.org/10.1016/0306-4549(94)90093-0.

2. A. GALPERIN and E. NISSAN, "Application of Heuristic Search Method for Generation of Fuel Reload Configurations," *Nucl. Sci. Eng.*, **99**, 343 (1988); https://doi.org/10.13182/NSE88-A23563.

3. A. GALPERIN, S. KIMHI, and E. NISSAN, "A Knowledge-Based System for Optimization of Fuel Reload Configurations," *Nucl. Sci. Eng.*, **102**, 43 (1989); https://doi.org/10.13182/NSE89-A23630.

4. D. E. GOLDBERG, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, Reading, New York (1989).

5. G. T. PARKS, "Multiobjective Pressurized Water Reactor Reload Core Design by Non-Dominated Genetic Algorithm Search," *Nucl. Sci. Eng.*, **124**, 178 (1996); https://doi.org/10.13182/NSE96-A24233.

6. A. A. M. MENESES, M. D. MACHADO, and R. SCHIRRU, "Particle Swarm Optimization Applied to Nuclear Reload Problem of a Pressurized Water Reactor," *Prog. Nucl. Energy*, **51**, *4*, 319 (2009); https://doi.org/10.1016/j.pnucene.2008.07.002.

7. S. JIANG et al., "Estimation of Distribution Algorithms for Nuclear Reactor Fuel Management Optimization," *Ann. Nucl. Energy*, **33**, 1039 (2006); https://doi.org/10.1016/j.anucene.2006.03.012.

8. S. MISHRA, R. S. MODAK, and S. GANESAN, "Optimization of Thorium Loading in Fresh Core of Indian PHWR by Evolutionary Algorithms," *Ann. Nucl. Energy*, **36**, 948 (2009); https://doi.org/10.1016/j.anucene.2009.03.003.

9. M. AGHAIE and S. M. MAHMOUDI, "A Novel Multi Objective Loading Pattern Optimization of PWRs by Gravitational Search Algorithm for WWER 1000 Core," *Prog. Nucl. Eng.*, **93**, 1 (2016); https://doi.org/10.1016/j.pnucene.2016.07.014.

10. L. MACHADO and R. SCHIRRU, "The Ant-Q Algorithm Applied to the Nuclear Reload Problem," *Ann. Nucl. Energy*, **29**, 1455 (2002); https://doi.org/10.1016/S0306-4549(01)00118-9.

11. A. THAKUR et al., "Performance of Estimation of Distribution Algorithm for Initial Core Loading Optimization of AHWR-LEU," *Ann. Nucl. Energy*, **96**, 230 (2016); https://doi.org/10.1016/j.anucene.2016.05.029.

12. R. K. SINHA and A. KAKODAR, "Design and Development of AHWR—The Indian Thorium Fueled Innovative Reactor," *Nucl. Eng. Design*, **236**, 683 (2006); https://doi.org/10.1016/j.nucengdes.2005.09.026.

13. A. THAKUR, B. SINGH, and P. D. KRISHNANI, "In-Core Fuel Management for AHWR-LEU," *Ann. Nucl. Energy*, **57**, 47 (2013); https://doi.org/10.1016/j.anucene.2013.01.034.

14. L. A. ZADEH, "Fuzzy Sets," *Inf. Control*, **8**, 338 (1965); https://doi.org/10.1016/S0019-9958(65)90241-X.

15. R. DAVID et al., "Gravitational Search Algorithm-Based Design of Fuzzy Control Systems with a Reduced Parametric Sensitivity," *Inf. Sci.*, **247**, 154 (2013); https://doi.org/10.1016/j.ins.2013.05.035.

16. H. G. KIM, S. H. CHANG, and B. H. LEE, "A Study on the Optimal Fuel Loading Pattern Design in Pressurized Water Reactor Using the Artificial Neural Network and the Fuzzy Rule Based System," *Proc. 4th Int. Topical Mtg. on Nuclear Thermal Hydraulics, Operator and Safety*, Taipei, Taiwan, April 6–8, 1994.

17. H. G. KIM, S. H. CHANG, and B. H. LEE, "Pressurized Water Reactor Core Parameter Prediction Using an Artificial Neural Network," *Nucl. Sci. Eng.*, **113**, 70 (1993); https://doi.org/10.13182/NSE93-A23994.

18. J. G. STEVENS, K. S. SMITH, and K. R. REMPE, "Optimization of Pressurized Water Reactor Shuffling by Simulated Annealing with Heuristics," *Nucl. Sci. Eng.*, **121**, 67 (1995); https://doi.org/10.13182/NSE121-67.

19. F.-S. SAEIDI-KHABISI and E. RASHEDI, "Fuzzy Gravitational Search Algorithm," *Proc. 2nd Int. Conf. on Computer and Knowledge Engineering (ICCKE)*, Iran, October 18–19, 2012.

20. M. AGHAIE and S. M. MAHMOUDI, "Multi Objective Loading Pattern Optimization of PWRs with Fuzzy Logic Controller Based Gravitational Search Algorithm," *Nucl. Eng. Des.*, **322**, 1 (2017); https://doi.org/10.1016/j.nucengdes.2017.06.036.

21. G. BOJADZIEV and M. BOJADZIEV, *Fuzzy Sets, Fuzzy Logics, Applications (Advances in Fuzzy*

⊗ANS

Systems—Applications and Theory Vol. 5), World Scientific Publication (1995).

22. O. A. ABDUL-REHMAN, M. MUNETOMO, and K. AKAMA, "An Adaptive Parameter Binary-Real Coded Genetic Algorithm for Constraint Optimization Problems: Performance Analysis and Estimation of Optimal Control Parameters," *Inf. Sci.*, **233**, 54 (2013); https://doi.org/10.1016/j.ins.2013.01.005.