Received: 21 February 2024 | Revised: 1 April 2024 | Accepted: 29 April 2024 | Published online: 8 May 2024

RESEARCH ARTICLE

Journal of Computational and Cognitive Engineering
2024, Vol. 3(4) 412–420
DOI: 10.47852/bonviewJCCE42022658

BON VIEW PUBLISHING

Evolutionary Search for Stochastic Optimization with Binary Choice Relations Under Fuzzy Modeling

Vyacheslav Irodov[1], **Serhii Dubrovskyi**[1,*], **Kostiantin Dudkin**[1], **Dmytro Chirin**[1] **and Yuliya Aldrich**[1]

[1]*Department of Information Technology, Dnipro Technological University, Ukraine*

Abstract: The article is devoted to presenting an approach to decision-making in fuzzy modeling of systems based on a limited number of experiments characterizing the system's behavior. An iterative algorithm is proposed for use, in which the functions of generation and selection of solutions with several branches of evolutionary search are successively implemented. The generation function is built, for the most part, regardless of the content of the task. The selection function is built using a selection procedure that is completely dependent on the problem to be solved. The resulting information is used to guide the search process, making it understandable for guided machine learning. The convergence of algorithms for finding optimal solutions in the presence of constraints in the form of inequalities and additional constraints in the form of binary relations is analyzed. The results of solving test problems of stochastic optimization are given. The described approach solves the problem of fuzzy modeling for decision-making based on a limited set of experimental data, which makes it possible to identify regularities and generalize them to evaluate the performance and accuracy of machine learning algorithms.

Keywords: evolutionary search, stochastic optimization, binary choice ratios, engineering, machine learning

1. Introduction

Stochastic optimization plays a significant role in the analysis, design, and operation of modern systems [1]. Stochastic optimization is a major branch of computational statistics. Stochastic optimization applies when there are noisy measurements of the criterion being optimized and/or there is an injected Monte Carlo randomness as part of the algorithm [2]. A significant number of works are devoted to stochastic optimization; first of all, these are the works of Jin et al. [3], Kizielewicz and Sałabun [4], and Paik and Mondal [5]. Uncertainties, risks, and disequilibrium are pervasive characteristics of modern socioeconomic, technological, and environmental systems involving interactions between humans, economics, technology, and nature. The systems are characterized by interdependencies, discontinuities, endogenous risks, and thresholds, requiring nonsmoothed quantile-based performance indicators, goals, and constraints for their analysis and planning [6, 7]. Evolutionary fuzzy systems are one of the greatest advances within the area of computational intelligence [2, 8]. They consist of evolutionary algorithms applied to the design of fuzzy systems [9]. Methods for modeling fuzzy systems have received significant development [10, 11]. The work of Chen [12] uses a developed modification of a genetic algorithm to optimize the operation of a neural network. A Solution Approach to Fuzzy Nonlinear Programming Problems was presented [13, 14], which is an example of solving numerical problems. The Approach to Solve the Fuzzy Nonlinear Unconstraint Optimization Problem was presented by Panigrahi et al. [15]; this paper investigates fuzzy nonlinear system equations using an optimization approach. In the paper by Iqbal et al. [16], a multi-objective non-linear programming problem in linear programming was proposed. Ensemble deep learning enabled multi-condition generative design of aerial building machine considering uncertainties was presented by Peng et al. [17]. The paper by Wang et al. [18] proposes one algorithm framework to solve multi-objective optimization problems. When setting the problem, real functions are used here to formulate criteria and restrictions, and binary selection relations are not used which narrows the possibilities of finding solutions to complex problems. A significant difference in the presented article's approach to decision-making is the use of binary choice relations. The use of binary choice relations for decision-making has a history in works from Devraj and Chen [19], Aizerman [20], Aizerman and Litvakov [21], and Sholomov and Yudin [22]. Decision-making in complex systems is achieved using self-organization methods developed in the works of Irodov et al. [23] and his followers [10]. In the works of Yudin and Sholomov [24] and other authors [25], computational methods of decision theory were considered, in which the problems of finding solutions are formulated in terms of binary relations; thus, the problems of nonlinear mathematical programming are transformed into problems of generalized mathematical programming. Methods of evolutionary search for solutions in problems with binary choice relations were developed [26]; a study of the convergence of evolutionary search algorithms for binary choice relations when applying the preference function in an iterative process is presented [27]. An evolutionary algorithm for multi-criteria optimization in

*__Corresponding author:__ Serhii Dubrovskyi, Department of Information Technology, Dnipro Technological University, Ukraine. Email: dubrovskyi_s@itstep.org

evolutionary search with a binary choice relation was presented [23], where the blocking function is used instead of the preference function. Liu et al. [28] proposed an algorithm for building a selection mechanism for decision-making in multi-criteria systems where there is a sample of fuzzy experimental results. The use of methods for finding solutions in intelligent systems, including the use of evolutionary search methods, genetic, and swarm algorithms, is widely presented, for example, in Fu et al. [29] and Zhang et al. [30]. Evolutionary search in stochastic optimization is used for final decision-making [31], but the study of the convergence of evolutionary search in stochastic optimization was not conducted.

The construction of evolutionary search algorithms for finding a solution using a fuzzy model of an object based on a limited set of experiments is of scientific and practical interest, which determines the purpose of the work performed.

The main goal of the article is to present the results of evolutionary algorithms for binary choice relations in stochastic optimization using the choice function in the form of preference. The study presents the results of the convergence of the evolutionary algorithm and provides examples of solving test problems in stochastic optimization. Using the described approach, the problem of fuzzy modeling for decision-making based on a limited set of experimental data is solved using the example of a film solar collector.

2. Methods of Stochastic Optimization with Binary Choice Relations

We will assume that the system is characterized by a set of parameters $x = \{x_1, x_2, \dots x_n\}$, and there are also initial parameters (functions, criteria) $z = \{z_1, z_2, \dots z_f\}$. We also assume that the fuzzy binary relation $\widetilde{R}_S$ with the membership function належності $\mu_{\widetilde{R}_S}(z, z)$ is known.

We assume that the known selection function $\Gamma(z)$ is such that

$$\Gamma(z(x_1))\Gamma(z(x_2)) \equiv z(x_1)\widetilde{R}_S z(x_2)$$

using the fuzzy binary relation $\widetilde{R}_S$ with the corresponding membership function $\mu_{\widetilde{R}_S}(z, z)$. It is necessary to find the dominant element x on the whole set Ω admissible parameters.

2.1. Ideal search algorithm

Significantly through M(iz) – mathematical definition of the parameter z – a set of outlet system functions (parameters).

The significant function of choice $S(x)$ is

$$S(x) = \left\{x \in X \middle| \forall y \in [X/S(x)], M(z(x))\widetilde{R}_S M(z(y))\right\} \tag{1}$$

Let us take into account that the choice relation $M(z(x))\tilde{R}_S M(z(y))$ is in a non-strict order, so it has the power of reflexivity, transitivity, and antisymmetric.

Apparently up to Iqbal et al. [16] and others [17], the evolutionary search looks like this:

$$X_{jk} = S(G(X_{jk-1})), j = \overline{1, N_B}, k = 1, 2, \dots \tag{2}$$

where

X_{jk} is the set of preferred solutions according to the binary choice relation $\widetilde{R}_S$ at the iterate step k for the branch j of evolutionary search.

$S(X)$ is the function of choice in the form (1).

$G(X)$ – generation function.

The generation function looks like

$$G(X) = X \cup G_H(X)$$

$$G_H(X) = \left\{y \in \Omega \middle| \exists x \in X, yR_G x, \mu_{R_G}(x, y) > 0\right\} \tag{3}$$

where relation R_G is the fuzzy generation relation with the dependency function:

$$\mu_{R_G}(x, y) : \Omega \times \Omega \to [0, 1]$$

It is significant that $R + S(x)$ is the upper strut of the relationship $\tilde{R}_S$ on the set of admissible parameters Ω:

$$R + S(x) = \left\{y\varepsilon\, \Omega \middle| y\, \widetilde{R}_S x\right\} \tag{4}$$

The following statement holds:

Statement 1. If $R_S^+(x)$ – the upper intersection with respect to relation $\tilde{R}_S$ has the property:

$$\forall x \neq x_0, mesR_S^+(x) > 0 \tag{5}$$

where x_0 is the $\tilde{R}_S$ – optimal solution on the set Ω – and the generating function satisfies the fact that if

$$x_H \in G_H(X) \text{ then}$$

$$\forall x \neq x_0, P\left\{x_H \in R_S^+(x)\right\}\delta > 0 \tag{6}$$

then in this case, for any $x \in \Omega, x \neq x_0$, there will be a number K such that for any $k \geq K$ and for all search branches $j = \overline{1, N_B}$ with probability 1, the requirement will be fulfilled $x_{jk} \subset R_S^+(x)$ which proves the convergence of the iterative process (2) with probability 1 to $\widetilde{R}_S$ – optimal solution for all branches of evolutionary search.

Various methods are known for generating solutions during evolutionary search. Search variables can include both continuous and discrete variables, including binary ones. Statement 1 defines sufficient conditions for the convergence of the evolutionary search to the most preferable solution. In this case, the essential point is that the generated solutions have such a property that there is always a finite, even if insignificant, probability of the newly generated solutions falling into the upper section of the choice relation.

2.2. A real search algorithm

In the selection function (1), we replace the mathematical expectation $M(z)$ on the sample mean value $\overline{z}$ which is calculated in the form

$$\overline{z} = 1/n_z \sum_{l=1}^{n_z} z_1$$

The selection function (1) takes the form

$$S(X) = \left\{x\varepsilon X \middle| \forall y \in \left[X \backslash S(X)\right], \overline{z}(x)\, \widetilde{R}_S\, \overline{z}(y)\right\} \tag{7}$$

Suppose that the solutions that passed the selection at some step of the iteration for all branches of the evolutionary search have the form $\{x_{lj}^i\}$ where i is the number of the variable value for the selected solutions l in the jth branch of the search, $j = \overline{1, N_B}$. N_B is a number of branches for evolutionary search. Average values for all selected solutions can be calculated as follows:

$$x_0^i = \frac{1}{N_B N_L} \sum_{j=1}^{N_\ell} \sum_{l=1}^{N_\Pi} x_{lj}^i \quad (8)$$

At the same time, the values of the empirical dispersion will be

$$\sigma_i^2 = \frac{1}{N_B N_L - 1} \sum_{j=1}^{N_B} \sum_{l=1}^{N_\Pi} (x_{lj}^i - x_0^i)^2 \quad (9)$$

The generation of new solutions at the next step of the iteration is performed with a normal distribution for each parameter x^i and centers in x_{lj}^i, $j = \overline{1, N_B}$ and variance σ_i^2. That is, the membership function μ_{R_G} for the fuzzy generation relation is the density function of the normal distribution:

$$\mu_{R_G}(y^i, x^i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{y^i - x^i}{\sigma_i}\right)^2\right] \quad (10)$$

Let us consider the case when the vector of the desired variables $x = \{x^1, x^2, \ldots, x^n\}$ consists of binary variables $x^i \in \{0, 1\}$. Let at some current step of the search iteration the $\widetilde{R}_S$ optimal solutions that have passed the selection procedure in all branches of the evolution of solutions have the form

$$\{x_{lj}^i\},\ i = 1, n;\ l = 1, N_S;\ j = 1, N_B.$$

Then we calculate estimates of the probability of the variable taking a value equal to 1 in the form

$$\overline{p}_i = \frac{1}{N_B N_S} \sum_{j=1}^{N_B} \sum_{l=1}^{N_S} x_{lj}^i$$

To generate new solutions at a new iteration step, the probability is calculated in the form, which is interpreted as the probability of the variable accepting the value 1:

$$p_i = \overline{p}_i \ \mathit{if}\ \delta \le \overline{p}_i \le 1 - \delta$$

$$\text{Or}\delta \ \mathit{if}\ \overline{p}_i < \delta$$

$$\text{Or } 1 - \delta \ \mathit{if}\ \overline{p}_i > 1 - \delta$$

Obviously, $(1 - p_i)$ is the probability of the variable accepting the value 0.

The described method for generating solutions for binary variables obviously satisfies the conditions of convergence of the evolutionary search.

3. Solution of Test Problems of Stochastic Optimization

We will present the results of solving test problems of stochastic optimization which practically demonstrate the capabilities of the described real solution search algorithm.

The first example is shown in Table 1, stochastic optimization of the Rosenbrock function:

$$F(x, y) = 0.25\eta[(1 - x)^2 + 100(y - x^2)^2] \rightarrow \min$$

where η is a random variable normally distributed with zero mean and variance of 1.

Table 1
Evolutionary search progress for stochastic optimization of the Rosenbrock function

Iteration step	Branch of evolution	x	y	F
1	1	6.566327	39.70555	1194.565
	2	5.72689	28.14048	2190.919
	3	5.201773	26.81788	23.46048
2	1	5.303123	26.79532	194.8043
	2	5.567961	30.72101	28.75454
	3	4.738819	22.32064	15.80767
37	1	1.343588	1.841613	0.233907
	2	2.349523	5.484628	1.956881
	3	1.237449	1.50857	0.082100
202	1	1.012769	1.023002	−0.06305
	2	1.033854	1.065072	−0.08682
	3	1.078498	1.173239	−0.05221

Table 1 shows a set of input system parameters x, y and dimensional parameters F.

The second example is shown in Table 2, stochastic minimization of the Rastrigin function:

$$F(x_i) = 0.25\eta[An + \sum_{i=1}^{n} (x_i^2 - A\cos(2\pi x_i))] \rightarrow \min$$

$A = 10$, $n = 2$, η − a is a random variable normally distributed with zero mean and variance of 1.

Table 2 shows a set of input system parameters $x1$, $y2$ and dimensional parameters F.

4. The Problem of Fuzzy Modeling of the Solar Collector

Mathematical modeling of a film-type solar collector is considered. The basis for this mathematical modeling is the results of an experimental study of the operation of a film-type solar collector. The results of the study of the work of the collector are presented in Table 3. There are seven dimensional parameters and three dimensionless parameters (complexes) that characterize the operating mode of the solar collector.

In Table 3, the first complex is $p_1 = \Delta T / T_{air}$ where ΔT is the temperature difference at the inlet and outlet of the solar collector and T_{air} is the air temperature.

The temperature complex p_1 characterizes the effect of air temperature on the difference between the initial and final water temperatures. The second complex characterizes the physical dimensions of the solar collector $p_2 = h^2/F$ where h is the distance between the translucent and absorbing surfaces and $F = 0.23m^2$ is the area of the translucent surface. The third dimensionless complex characterizes the efficiency of the solar collector $p_3 = c_w M_w \Delta T / (q_{sun} F)$, where q_{sun} is the solar radiation heat flux density, M_w is the mass flow rate of water through the collector, and c_w is the mass heat capacity of

Table 2
Evolutionary search progress for stochastic optimization of the Rastrigin function

Iteration step	Branch of evolution	x_1	y_2	F
1	1	2.15053	2.87881	19.7966
	1	3.18624	2.09815	22.4829
	1	1.89483	3.22277	24.3964
	2	2.82108	2.13555	21.6122
	2	3.84588	2.05549	23.9398
	2	3.05669	3.19328	26.6383
	3	2.92202	1.11057	13.2241
	3	2.22859	−0.936689	15.2639
	3	2.26403	1.98179	19.9844
10	1	−0.9237613	1.01625	3.0167
	1	0.10237	1.02406	3.2196
	1	−0.1301714	−1.040702	4.57903
	2	−0.0487178	1.09186	3.26372
	2	2.03096	0.96108	5.56181
	2	2.01282	−1.976363	8.12123
	3	1.89749	0.00074	5.57147
	3	−1.004367	−1.137878	5.79307
	3	−1.033895	−0.170471	6.54668
22	1	−0.009145023	−0.000178351	−0.0226898
	1	0.00617	0.00519	0.00572
	1	0.00167	0.01537	0.00653
	2	0.0012	−0.01025081	−0.00042560
	2	−5.4419e-005	−0.01072413	0.00035
	2	−0.00498262	0.00031	0.00108
	3	−0.001211888	−0.000880654	−0.05330217
	3	0.00314	0.00072	−0.02399012
	3	−0.001316805	−0.00627928	−0.00571147

Table 3
The results of research on the operation of a film-type solar collector

Dimensional parameters						Dimensionless complexes		
T_{inlet}, K	T_{outlet}	T_{air}, K	h, sm	q_{sun}	$M_w, g/s$	p_1	p_2	p_3
297.5	300.7	303.97	2	99.999	6.67	0.103326	0.001739	0.388186
297.3	301.4	303.71	2	98.862	6.67	0.133507	0.001739	0.503084
297.5	301.6	304.33	1	88.161	6.67	0.130865	0.000435	0.564148
297.7	302.3	304.16	1	100.491	6.67	0.147625	0.000435	0.555286
299.0	302.5	301.87	1	107.467	6.67	0.121233	0.000435	0.395074
299.2	303.0	302.39	1	97.374	6.67	0.129296	0.000435	0.473398
299.9	304.8	302.22	1	100.465	5.0	0.167693	0.000435	0.443518
301.1	305.2	302.3	1	101.282	5.0	0.174061	0.000435	0.457897
300.5	305.7	303.09	2	91.177	5.0	0.172815	0.001739	0.518619
300.7	305.9	303.71	2	98.862	5.0	0.169326	0.001739	0.478304
300.8	305.1	301.26	2	74.74	5.0	0.152159	0.001739	0.523173
300.1	307.8	303.36	2	111.016	3.85	0.253623	0.001739	0.485653
300.2	307.4	301.78	2	89.27	3.85	0.250174	0.001739	0.564739
300.3	307.8	300.75	2	94.154	3.85	0.27027	0.001739	0.557755
300.4	307.4	303.89	2	61.763	3.85	0.226611	0.001739	0.793579
300.4	306.6	303.89	1	83.502	3.85	0.200712	0.000435	0.519895
300.1	307.8	302.83	1	86.186	3.85	0.258129	0.000435	0.625568
300.7	306.9	301.52	1	113.943	3.85	0.217391	0.000435	0.381
301.2	307.6	302.04	2	112.004	3.85	0.220386	0.001739	0.400099
301.3	305.9	302.13	2	76.28	3.85	0.157913	0.001739	0.422248

water. For the p_3 complex, the following empirical dependence was obtained:

$$p_3 = 0.53 - 0.295p_1 + 0.0027(1 - p_2) + 3.5p_1^2 - 0.23(1 - p_2)^2$$

As can be seen from the analysis of the experimental results of a film-type solar collector, the optimal operating modes from the point of view of the maximum criteria p_1 and p_3 are somewhat different. Therefore, it is desirable to find a compromise mode of operation of the reservoir, which emphasizes the feasibility of this scientific work.

Based on the results of experiments, it is difficult to determine the most preferable mode of operation of the collector, especially if the task is to find the most preferable mode not only among the set of experiments but also if you expand the search to the entire set of permissible parameters. To solve the problem, fuzzy modeling of the reservoir operation was used, functions for selecting the most preferable solutions were constructed, and the maximum of this selection function was found on a set of acceptable parameters.

For expert evaluation, the rating scale used bit □ {0; 0.3; 0.4; 0.5; 0.6; 0.7; 1.0}, which make sense: {much worse; worse; slightly worse; comparable; slightly better; better; much better}. The correspondence matrix for the array of experimental data is shown in Tables 4 and 5.

Table 4
Correspondence matrix for the array of experimental data 1

DATA 0.5, 0.4, 0.3, 0.4, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.3
DATA 0.6, 0.5, 0.6, 0.4, 0.4, 0.4, 0.0, 0.3, 0.3, 0.3, 0.3, 0.3
DATA 0.5, 0.4, 0.5, 0.3, 0.3, 0.4, 0.0, 0.3, 0.3, 0.3, 0.3, 0.3
DATA 0.6, 0.6, 0.6, 0.5, 0.4, 0.5, 0.3, 0.3, 0.3, 0.4, 0.4, 0.4
DATA 0.7, 0.7, 0.7, 0.6, 0.5, 0.6, 0.4, 0.4, 0.4, 0.4, 0.5, 0.5
DATA 0.6, 0.6, 0.6, 0.5, 0.5, 0.5, 0.3, 0.3, 0.4, 0.4, 0.5, 0.4
DATA 1.0, 1.0, 1.0, 0.6, 0.5, 0.6, 0.5, 0.5, 0.5, 0.5, 0.6, 0.6
DATA 0.7, 0.7, 0.7, 0.6, 0.6, 0.6, 0.6, 0.6, 0.5, 0.5, 0.5, 0.5
DATA 0.7, 0.7, 0.7, 0.6, 0.6, 0.6, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5
DATA 0.7, 0.7, 0.7, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5
DATA 0.7, 0.7, 0.7, 0.6, 0.6, 0.6, 0.5, 0.5, 0.5, 0.5, 0.3, 0.4
DATA 0.7, 0.7, 0.7, 0.6, 0.6, 0.6, 0.5, 0.5, 0.5, 0.5, 0.4, 0.4

Table 5
Correspondence matrix for the array of experimental data 2

DATA 0.5, 0.5, 0.6, 0.6, 0.5, 0.7, 0.3, 0.3
DATA 0.4, 0.5, 0.6, 0.6, 0.5, 0.6, 0.3, 0.3
DATA 0.3, 0.3, 0.5, 0.3, 0.5, 0.5, 0.3, 0.3
DATA 0.3, 0.3, 0.3, 0.5, 0.3, 0.3, 0.3, 0.3
DATA 0.4, 0.5, 0.6, 0.6, 0.5, 0.6, 0.4, 0.4
DATA 0.7, 0.7, 0.6, 0.6, 0.6, 0.5, 0.5, 0.5
DATA 0.7, 0.7, 0.7, 0.7, 0.7, 0.7, 0.5, 0.5
DATA 0.7, 0.7, 0.5, 0.7, 0.5, 0.5, 0.5, 0.5

Table 3 shows T_{inlet}, K, T_{outlet}, T_{air}, K, h, sm, $q_{sun}, mW/sm^2$, $M_w, g/s$ are dimensional parameters and $c_w = 4183$ J/kg/K; p_1, p_2, p_3 are dimensionless parameters (complexes).

There are presented results with choice function in the form (11)

$$\Gamma(x) = \prod_{i=1}^{5} \left(1 + a_{1i} \cdot (a_{2i} - r_i)^2\right) \tag{11}$$

where a_{1i} and a_{2i} are the choice function parameters:

$$\begin{aligned} &r_1 = x_1^1 - x_2^1;\ r_2 = x_1^2 - x_2^2; r_3 = x_1^3 - x_2^3; \\ &r_4 = x_1^4 - x_2^4;\ r_5 = x_1^5 - x_2^5; \end{aligned} \tag{12}$$

$$\Gamma(x_1) \geq \Gamma(x_2) \equiv x_1 \cdot \tilde{R}_s \cdot x_2 \tag{13}$$

Parameters a_{1i} and a_{2i} were obtained after evolutionary search of the choice function for array 1 of experimental data and for array 2 of experimental data.

Step 1. Input of a table of experimental data with dimensional parameters. Transition to a set of dimensionless parameters p_1, p_2, p_3 for the entire set of experiments, number of experiments $N_T = 20$. We divide the entire set of experiments into two arrays: (1) training sequence array and (2) testing sequence array. The number of experiments for training sequence array $N_o = 12$, for the array *DATA* 1, 3, 5, 7, 8, 9, 11, 13, 15, 16, 17, 19. The number of experiments for testing sequence array $N_{TP} = 8$ for the array *DATA* 2, 4, 6, 10, 12, 14, 18, 20.

Step 2. Using expert assessments, two matrices of paired comparisons are constructed: the matrix for array 1 of experimental data for training sequence array and the matrix for array 2 of experimental data for testing sequence array.

Step 3. The parameters of the evolution search are selected. The number of search parameters $n = 10$; the number of generated solutions for one branch of evolution and for one iteration step $Ne = 10$; the number of selected solutions $NS = 1$; the branch number $j = \overline{1, N_B}$, $N_B = 3$; and the number of search repetitions at a common iteration step without changing random parameters is 5.

$$For\ i = 1 to n(\sigma_1 = 0.1 : x^i_{midl} = 0 : next\ i)$$

Step 4. Generation of initial values of the required parameters for all generated solutions for all branches of evolution. Preparatory calculations are being carried out.

$$\begin{aligned} &For\ k = 1\ to\ N_E(for\ i = 1\ to\ n(\ jf = mn(k, iv); \\ &yel = rnd(1);\ w1 = yel * jd;\ lt = \mathrm{int}(w1) + 1; \end{aligned}$$

$$\begin{aligned} &jg = mn(lt, iv);\ d = a(i, jg);\ x_T(i) = d; \\ &ys = 0;\ for\ il = 1\ to\ 12(\ z = rnd(1); \end{aligned}$$

$$\begin{aligned} &ys = ys + z;\ next\ il)\ ys = ys - 6; \\ &v1 = d + \sigma(i) * ys;\ a(i, jf) = v1;\ next\ i)\ next\ k). \end{aligned}$$

Step 5. Calculation of objective functions for all possible solutions for a given branch of evolution:

$$\begin{aligned} &for\ k = jn\ to\ N_E(j = mn(k, iv) : \\ &for\ i = 1\ to\ 5(\ a_1(i) = a(i, j) : \\ &i_2 = i + 5 : a_2(i) = a(i_2, j) : next\ i) \end{aligned}$$

$$\begin{aligned} &s = 0 : for\ jj = 1\ to\ N_O(for\ ii = 1\ to\ N_O(j_1 = mo(jj) : \\ &x_1 = p_3(j_1) : j_2 = mo(ii) : x_2 = p_3(j_2) \end{aligned}$$

$$r(1) = x_1 - x_2 : r(2) = {x_1}^2 - {x_2}^2 : r(3) = {x_1}^3 - {x_2}^3 :$$
$$r(4) = {x_1}^4 - {x_2}^4 : r(5) = {x_1}^5 - {x_2}^5 : s = 1$$

$$for\ i = 1\ to\ 5(y = a_1(i) \cdot (a_2(i) - r(i) \cdot (a_2(i) - r(i)) :$$
$$s = s \cdot y\ next\ i)\ ej(jj, ii) = s\ next\ ii)\ next\ jj)$$

$$next\ k)\ for\ k = jn\ to\ N_E(\ j = mn(k, iv) :$$
$$s_1 = 0\ for\ jj = 1\ to\ N_O(for\ ii = 1\ to\ N_O$$

$$(ra = xe(jj, ii) - ey(jj, ii) : s_1 = s_1 + abs(ra) :$$
$$next\ ii)\ next\ jj)\ e_1(j) = s_1/N_O/N_O\ next\ k)$$

Step 6. Selection of the most preferable solutions:

$$i_1 = N_E - 1\ for\ l = 1\ to$$
$$N_S(\ for\ lt = 1\ to\ i_1\ (\ j = N_E - lt + 1 :$$
$$k - j - 1 : jf = mn(j, iv) : kf = mn(k, iv)$$

$$if\ e_1(kf) \leq e_1(jf)\ goto\ z : mn(j, iv) = kf :$$
$$mn(k, iv) = jf : z = 1 : next\ lt)\ next\ l)\ next\ iv)$$

Step 7. Print results for iteration step:

$$kit = kit + 1\ prnt\ kit :$$
$$for\ iv = 1\ to\ N_B(\ for\ k = 1\ to\ N_S(\ jf = mn(k, iv) :$$
$$for\ i = 1\ to\ n\ (prnt\ (a(i, jf), e1(jf))$$

Step 8. The calculation of parameters for the generation of new solutions at the new iteration step:

$$for\ i = 1\ to\ n\ (sr = 0 : for\ iv = 1\ to\ N_B(for\ k = 1\ to$$
$$N_S(j = mn(k, iv) : sr = sr + a(i, j)$$

$$next\ k)\ next\ iv)\ sr = sr/N_B/N_S : xsr(i) = sr :$$
$$d = 0 :\ for\ iv = 1\ to\ N_B(\ for\ k = 1\ to\ N_S)$$

$$(j = mn(k, iv) : ra = a(i, j) - sr : d = d + ra \cdot ra :$$
$$next\ k)\ next\ iv)\ ra = d/(N_S \cdot N_B - 1)$$

$$sg(i) = sqr(ra) : next\ i)\ jd = N_S :$$
$$zzz = 1\ goto\ step\ 4$$

Step 9. Print results.

Table 6 demonstrates the evolutionary search iteration step. There is a reduction in the error in constructing the choice function model for the training data array, as well as the error for the testing data array. The error for the verification data differs slightly from the error for the training data set, which indicates the adequacy of the constructed model.

Table 6
Evolutionary search of the selection function

Evolutionary search iteration step	Error for training sequence matrix 1	Error for check sequence matrix 2
1	0.4981	
2	0.3449	
6	0.2730	
8	0.1801	
12	0.1319	0.1387
20	0.1353	

The results of specific values of parameters a_{1i},a_{2i} $i = \overline{1,5}$ were as follows:
1) I; A1(I); A2(I) = 1, 0.1496397–1.427655
2) I; A1(I); A2(I) = 2, 1.62397−0.3657375
3) I; A1(I); A2(I) = 3, 0.4951336−0.2666837
4) I; A1(I); A2(I) = 4, 0.0420600–0.0120022
5) I; A1(I); A2(I) = 5, 0.0218973−2.013484

The choice function in the form (11) with specific values of parameters a_{1i},a_{2i}, $i = \overline{1,5}$ was used to solve the problem of generalized mathematical programming: to find maximum MAX$\mu_{\tilde{R}_S}(x,y)$ of choice function with restrictions, $0.05 <= x_1 =< 0.25$; $0.001 <= x_2 =< 0.004$; $0.15 <= x_3 <= 0.8$.

4.1. Algorithm maximum

Step 1. At this step, the permissible limits for changing parameters, the number of generated solutions, the number of selected most preferable solutions, the number of branches of the evolution of solutions, the standard deviation of solutions from the values obtained as a result of a physical experiment, and the number of calculations of the selection function to obtain estimates from values for random deviations are determined:

$$n = 3 : for\ i = 1\ n(\ sg(i) = 0.05 : x_{SR}(i) = 0 :$$
$$next\ i) : \min(1) = 0.05 : \max(1) = 0.25 : \min(2) = 0.001 :$$
$$\max(2) = 0.004 : \min(3) = 0.15 : \max(3) = 0.8 :$$
$$N_E = 10 : N_S = 3 : N_B = 3 : svg = 0.05 : N_{vich} = 10$$

Step 2. Generation of initial values of the required parameters for all generated solutions for all branches of evolution:

$$l = 0 :\ for\ iv = 1\ to\ N_B(\ for\ k = 1\ to\ N_S(l = l + 1 :$$
$$mn(k, iv) = l : next\ k)\ next\ iv)\ for\ i = 1\ to\ n$$

$$for\ iv = 1\ to\ N_B(\ j = mn(1, iv) : a(i, j) = xsr(i) :$$
$$next\ iv)\ xt(i) = xsr(i) : next\ i)$$

Step 3. Preparatory calculations are being carried out. For the initial values of all parameters of the search for solutions, deviations are calculated using the standard function for generating random variables.

$$jd = 0 : jn = jd + 1 : for\ kp = 1\ to\ N_{REP}(\ for\ iv = 1\ to\ N_B(\ for\ k = jn\ to\ N_E(\ for\ i = 1\ to\ n\ )$$

$$(jf = mn(k, iv) : yel = rnd(1) : w_1 = yel \cdot jd : lt = \text{int}(w1) + 1 : jg = mn(lt, iv) : d = a(i, jg)$$

$$xt(i) = d : ys = 0 : for\ il = 1\ to\ 12\ (z = rnd(1) : ys = ys + z : next\ il)\ ys = ys - 6$$

$$v_1 = d + sg(i) \cdot ys : if\ v_1 < \min(i)\ then\ v_1 = \min(i) : if\ v_1 > \max(i)\ then\ v_1 = \max(i)$$

$$a(i, jf) = v_1 : next\ i)\ z_1 = a(1, jf) : z_2 = a(2, jf) : z_e = 0.51 - 0.295 \cdot z_1 + 0.0027 \cdot (1 - z_2) + 3.5 \cdot z_1 \cdot z_1$$

$$z_3 = z_3 - 0.23 \cdot (1 - z_2) \cdot (1 - z_2) : if\ z_3 < 0.01\ then\ z_3 = 0.01\ \ if\ z_3 > 0.7\ then\ z_3 = 0.7$$

$$a(3, jf) = z_3 : next\ k)\ for\ k = jn\ toN_E(j = mn(k, iv) : x_1 = a(3, j) : next\ k)$$

Step 4. Calculation of objective functions for all possible solutions for a given branch of evolution. In this case, previously found parameter values for the choice function are used.

$$sum = 0 : for\ ivich = 1\ to\ N_{vich}(for\ il = 1\ to\ 12\ (z = rnd(1) : ys = ys + z : next\ il)$$

$$ys = ys - 6 : v_1 = x_1 + sgv \cdot ys : if\ v_1 < 0.01\ then\ v_1 = 0.01 : if\ v_1 > 0.8\ then\ v_1 = 0.8 : x_2 = v_1$$

$$r(1) = x_1 - x_2 : r(2) = {x_1}^2 - {x_2}^2 : r(3) = {x_1}^3 - {x_2}^3 : r(4) = {x_1}^4 - {x_2}^4 : r(5) = {x_1}^5 - {x_2}^5$$

$$s_1 = 1 : for\ i = 1\ to\ 5(\ yy = 1 + a_1(i) \cdot (a_2(i) - r(i)) \cdot (a_2(i) - r(i)) : s_1 = s_1 \cdot yy : next\ i)$$

$$sum = sum + s_1 : next\ ivich)\ sum = sum/N_{vich} : e_1(j) = sum : next\ k)$$

Step 5. Selection of the most preferable solutions.

Here, the most preferable solutions are directly selected based on the values of the choice function; those solutions that have large values of the choice function are selected:

$$i_1 = N_E - 1\ for\ l = 1\ to\ N_S(\ for\ lt = 1\ to\ i1\ (\ j = N_E - lt + 1 : k = j - 1 : jf = mn(j, iv) : kf = mn(k, iv))$$

$$if\ e_1(kf) \geq e_1(jf)\ goto\ z : mn(j, iv) = kf : mn(k, iv) = jf : z = 1 : next\ lt)\ next\ l)\ next\ iv)$$

Step 6. The calculation of parameters for the generation of new solutions at new iteration step. Here, the average values for each parameter among the selected most preferable solutions for all branches of evolution and the standard deviations of these parameters are calculated.

$$for\ i = 1\ to\ n\ (sr = 0 : for\ iv = 1\ to\ N_B(for\ k = 1\ to\ N_S(j = mn(k, iv) : sr = sr + a(i, j)$$

$$next\ k)\ next\ iv)\ sr = sr/N_B/N_S : xsr(i) = sr : d = 0 :\ for\ iv = 1\ to\ N_B(\ for\ k = 1\ to\ N_S$$

$$(j = mn(k, iv) : ra = a(i, j) - sr : d = d + ra \cdot ra : next\ k)\ next\ iv)\ ra = d/(N_S \cdot N_B - 1)$$

$$sg(i) = sqr(ra) : next\ i)\ jd = N_S : zzz = 1\ goto\ step\ 4$$

Table 7 shows the search process in three branches of evolution. It is clearly seen that in just 10 iteration steps, values of the maximum of the choice function in all branches of evolution were obtained that were quite close to each other, that is, the convergence of the evolutionary search was obtained. And the very small number of steps of evolutionary search clearly indicates the effectiveness of its use for solving similar problems.

Table 7
Evolutionary search for the maximum of the choice function

Evolutionary search step	Maximum selection function $\text{MAX}\mu_{\tilde{R}_S}(x,y)$ branch 1 of evolution	Maximum selection function $\text{MAX}\mu_{\tilde{R}_S}(x,y)$ branch 2 of evolution	Maximum selection function $\text{MAX}\mu_{\tilde{R}_S}(x,y)$ branch 3 of evolution
1	0.6171489	0.6229391	0.6348273
2	0.6334881	0.6234488	0.6348273
3	0.6334881	0.6244889	0.6348273
...			
10	0.6367123	0.6341401	0.6348273

Thus, the most preferable parameters for the solar collector have been found as shown in Table 8: $p_1 = 0.183$; $p_2 = 0.002$; $p_3 = 0.347$.

Table 8
Evolutionary search results when finding the maximum of the selection function

Branch of evolution	Parameter x_1	Parameter x_2	Parameter x_3
Branch 1	0.1827321	0.0026508	0.3468732
Branch 2	0.176442	0.0017470	0.340409
Branch 3	0.1887753	0.0022285	0.3527556

This solution corresponds to the maximum of the selection function, that is, the best not only among the set of experimental data but also in the entire set of varied parameters that differed from the set of Table 3 with a random standard deviation $\sigma_i = 0.05$ from the normal distribution.

The final result is selected when the values of the selected selection functions and the values of the parameters of the selected solutions in all branches of evolution are sufficiently close to each other. In this case, this is done.

5. Conclusion

The article considers approaches to decision-making in fuzzy modeling of systems based on a limited number of experiments for some generalized mathematical programming problems. Multi-criteria optimization is presented as optimization using binary relations.

A modification of the method [32] using a fuzzy object comparison scale is proposed.

This article presents the results of evolutionary algorithms for binary choice relations in stochastic optimization using the choice function in the form of preference. The convergence of the evolutionary algorithm is investigated, and the results of solving test problems for stochastic optimization are given.

The results of the calculations prove that the algorithm of evolutionary search has a sufficiently good performance in solving problems of fuzzy modeling. This solution corresponds to the maximum of the choice function and is the best not only among the set of experimental data but also in the entire set of varied parameters that differed from the set of physical experiment data.

The results of the research show that the proposed approach to building evolutionary search algorithms allows solving stochastic optimization problems and applying this algorithm to build a choice function in case of fuzzy modeling of solar collector operation, including for finding the maximum of this choice function.

It is important that the algorithm can be built to a large extent independent of the content of the problem to generate new solutions, which makes it a universal tool.

Recommendations

The use of evolutionary search for stochastic optimization in finding the maximum of the choice function makes it possible to find the optimal solution not only among the performed experiments but also to expand the search space to the set of all permissible parameters, including those parameters for which there has not yet been an experimental study.

In the future, it is advisable to investigate the interaction of various aspects of fuzzy choice on the final decision on the entire set of permissible parameters and not just on the set of experimental results. It is of interest to solve the following modification of the problem considered in this article. Let it be required to find a solution $x \in \Omega$ and for all $y \in \Omega$ so that $\Gamma_1(x_1) \geq \Gamma_1(x_2)$ and also it is fulfilled $\Gamma_2(x_1) \geq \Gamma_2(x_2)$. Let us create a new binary relation $\widetilde{R}_S$ in the form

$$x_1\widetilde{R}_S x_2 \equiv [\Gamma_1(z(x_1)) > \Gamma_1(z(x_2))] \vee [\Gamma_1(z(x_1)) = \Gamma_1(z(x_2))] \wedge [\Gamma_2(z(x_1)) \geq \Gamma_2(z(x_2))]$$

It seems possible to use the evolutionary search algorithm to find a solution to such a problem. This will be discussed in subsequent works. In general, as experimental studies of a film-type solar collector show, it has acceptable efficiency values at low temperatures of water in the film and lower efficiency values at elevated temperatures of water in the collector film. Therefore, there is a natural desire to use film solar collectors together with a heat pump, so that the film collectors operate at low water temperatures in the collector, and the heat pump ensures that the temperature of the supplied water is brought to the desired value. There are a number of scientific results that study the joint operation of solar collectors and heat pumps. It seems most appropriate to conduct further research in this direction.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Author Contribution Statement

Vyacheslav Irodov: Conceptualization, Methodology, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing. **Serhii Dubrovskyi:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Kostiantin Dudkin:** Validation, Investigation, Resources. **Dmytro Chirin:** Validation, Resources. **Yuliya Aldrich:** Validation, Formal analysis.

References

[1] Chen, T., Sun, Y., Xiao, Q., & Yin, W. (2021). A single-timescale method for stochastic bilevel optimization. *arXiv Preprint: 2102.04671*.

[2] Feng, J., Liu, S., Yang, Y., Zheng, J., & Liu, J. (2023). An adaptive tradeoff evolutionary algorithm with composite differential evolution for constrained multi-objective optimization. *Swarm and Evolutionary Computation*, *83*, 101386. https://doi.org/10.1016/j.swevo.2023.101386

[3] Jin, W., Hu, Z., & Chan, C. (2017). An innovative genetic algorithms-based inexact non-linear programming problem solving method. *Journal of Environmental Protection*, *8*(3), 231–249. https://doi.org/10.4236/jep.2017.83018

[4] Kizielewicz, B., & Sałabun, W. (2020). A new approach to identifying a multi-criteria decision model based on stochastic optimization techniques. *Symmetry*, *12*(9), 1551. https://doi.org/10.3390/sym12091551

[5] Paik, B., & Mondal, S. K. (2023). Scoring rule and its application in intuitionistic fuzzy parameterized soft set-based decision-making problem. *Journal of Ambient Intelligence and Humanized Computing*, *14*(10), 14209–14224. https://doi.org/10.1007/s12652-023-04658-x

[6] Ermoliev, Y., Zagorodny, A. G., Bogdanov, V. L., Ermolieva, T. E., Havlik, P., Rovenskaya, E., . . . , & Obersteiner, M. (2022). Robust food–energy–water–environmental security management: Stochastic quasigradient procedure for linkage of distributed optimization models under asymmetric information and uncertainty. *Cybernetics and Systems Analysis*, *58*(1), 45–57. https://doi.org/10.1007/s10559-022-00434-5

[7] Ermolieva, T., Ermoliev, Y., Obersteiner, M., & Rovenskaya, E. (2021). Two-stage nonsmooth stochastic optimization and iterative stochastic quasigradient procedures for robust estimation, machine learning and decision making. In F. S. Roberts & I. A. Sheremet (Eds.), *Resilience in the digital age* (pp. 45–74). Springer. https://doi.org/10.1007/978-3-030-70370-7_4

[8] Fernandez, A., Herrera, F., Cordon, O., del Jesus, M. J., & Marcelloni, F. (2019). Evolutionary fuzzy systems for

explainable artificial intelligence: Why, when, what for, and where to? *IEEE Computational Intelligence Magazine*, *14*(1), 69–81. https://doi.org/10.1109/MCI.2018.2881645

[9] Yang, Y., Liu, J., Tan, S., & Wang, H. (2019). A multi-objective differential evolutionary algorithm for constrained multi-objective optimization problems with low feasible ratio. *Applied Soft Computing*, *80*, 42–56. https://doi.org/10.1016/j.asoc.2019.02.041

[10] Sharaf, I. M. (2023). A cognitive-based similarity measure for decision-making with spherical fuzzy information. *Journal of Computational and Cognitive Engineering*, *2*(4), 331–342. https://doi.org/10.47852/bonviewJCCE3202479

[11] Zedam, L., Ullah, K., Sajjad, H. M., Hussain, A., & Parveen, A. (2023). Maclaurin symmetric mean aggregation operators based on spherical fuzzy information and application to decision-making. *Journal of Computational and Cognitive Engineering*, *2*(4), 266–277. https://doi.org/10.47852/bonviewJCCE3202551

[12] Chen, Z. (2022). Research on internet security situation awareness prediction technology based on improved RBF neural network algorithm. *Journal of Computational and Cognitive Engineering*, *1*(3), 103–108. https://doi.org/10.47852/bonviewJCCE149145205514

[13] Umamaheswari, P., & Ganesan, K. (2017). A solution approach to fuzzy nonlinear programming problems. *International Journal of Pure and Applied Mathematics*, *113*(13), 291–300.

[14] Ünver, M., Olgun, M., & Türkarslan, E. (2022). Cosine and cotangent similarity measures based on Choquet integral for spherical fuzzy sets and applications to pattern recognition. *Journal of Computational and Cognitive Engineering*, *1*(1), 21–31. https://doi.org/10.47852/bonviewJCCE2022010105

[15] Panigrahi, K., Nayak, S., & Priyadarshini, S. (2022). A new approach to solve the fuzzy nonlinear unconstraint optimization problem. *arXiv Preprint: 2206.00453*.

[16] Iqbal, S., Yaqoob, N., & Gulistan, M. (2023). Multi-objective non-linear programming problems in linear diophantine fuzzy environment. *Axioms*, *12*(11), 1048. https://doi.org/10.3390/axioms12111048

[17] Peng, F., Lv, L., Chen, W., & Wang, J. (2023). A projection-based evolutionary algorithm for multi-objective and many-objective optimization. *Processes*, *11*(5), 1564. https://doi.org/10.3390/pr11051564

[18] Wang, J., Chen, K., Yang, H., & Zhang, L. (2024). Ensemble deep learning enabled multi-condition generative design of aerial building machine considering uncertainties. *Automation in Construction*, *157*, 105134. https://doi.org/10.1016/j.autcon.2023.105134

[19] Devraj, A. M., & Chen, J. (2019). Stochastic variance reduced primal dual algorithms for empirical composition optimization. *arXiv Preprint: 1907.09150*.

[20] Aizerman, M. A. (1984). Some new problems in the general theory of choice (one line of research). *Avtomatika i Telemekhanika*, *45*(9), 5–43.

[21] Aizerman, M. A., & Litvakov, B. M. (1988). On some extensions of the option choice theory (findamentals of the pseudocriterion theory). *Avtomatika i Telemekhanika*, *49*(3), 92–105.

[22] Sholomov, L. A., & Yudin, D. B. (1986). Design of multistep choice schemes. *Avtomatika i Telemekhanika*, (10), 115–126.

[23] Irodov, V. F., Barsuk, R. V., & Chornomorets, H. Y. (2020). Multiobjective optimization at evolutionary search with binary choice relations. *Cybernetics and Systems Analysis*, *56*(3), 449–454. https://doi.org/10.1007/s10559-020-00260-7

[24] Yudin, D. B., & Sholomov, L. A. (1985). Multistep schemes of generalized mathematical programming and the choice function. *Doklady Akademii Nauk SSSR*, *282*(5), 1066–1069.

[25] Kolbin, V. V. (2014). Generalized mathematical programming as a decision model. *Applied Mathematical Sciences*, *8*(70), 3469–3476. http://doi.org/10.12988/ams.2014.44231

[26] Wang, B. C., Shui, Z. Y., Feng, Y., & Ma, Z. (2022). Evolutionary algorithm with dynamic population size for constrained multi-objective optimization. *Swarm and Evolutionary Computation*, *73*, 101104. https://doi.org/10.1016/j.swevo.2022.101104

[27] Yang, J., Ji, K., & Liang, Y. (2021). Provably faster algorithms for bilevel optimization. *arXiv Preprint: 2106.04692*.

[28] Liu, B., Zhang, Q., Fernández, F. V., & Gielen, G. G. E. (2013). An efficient evolutionary algorithm for chance-constrained bi-objective stochastic optimization. *IEEE Transactions on Evolutionary Computation*, *17*(6), 786–796. https://doi.org/10.1109/TEVC.2013.2244898

[29] Fu, Y., Hou, Y., Wang, Z., Wu, X., Gao, K., & Wang, L. (2021). Distributed scheduling problems in intelligent manufacturing systems. *Tsinghua Science and Technology*, *26*(5), 625–645. https://doi.org/10.26599/TST.2021.9010009

[30] Zhang, S., Xu, Y., Yu, Z., Zhang, W., & Yu, D. (2017). Combining extended imperialist competitive algorithm with a genetic algorithm to solve the distributed integration of process planning and scheduling problem. *Mathematical Problems in Engineering*, *2017*(1), 9628935. https://doi.org/10.1155/2017/9628935

[31] Irodov, V. F., Shaptala, M. V., Dudkin, K. V., Shaptala, D. E., & Chirin, D. A. (2022). Decision-making at evolutionary search during limited number of fuzzy experiments with multiple criteria. *Radio Electronics, Computer Science, Control*, *1*(1), 167. https://doi.org/10.15588/1607-3274-2022-1-17

[32] Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, *1*(1), 83–98. https://doi.org/10.1504/IJSSCI.2008.017590

How to Cite: Irodov, V., Dubrovskyi, S., Dudkin, K., Chirin, D., & Aldrich, Y. (2024). Evolutionary Search for Stochastic Optimization with Binary Choice Relations Under Fuzzy Modeling. *Journal of Computational and Cognitive Engineering*, *3*(4), 412–420. https://doi.org/10.47852/bonviewJCCE42022658