Journal of Algorithms and Computation

journal homepage: http://jac.ut.ac.ir

Solving a non-linear optimization problem in the presence of Yager-FRE constraints

Amin Ghodousian*[1], Abolfazl Javan†[2] and Asieh Khoshnood‡[3]

[1]Department of Engineering Science, College of Engineering, University of Tehran.
[2,3]Department of Algorithms and Computation, University of Tehran.

ABSTRACT

Yager family of t-norms is a parametric family of continuous nilpotent t-norms which is also one of the most frequently applied one. This family of t-norms is strictly increasing in its parameter and covers the whole spectrum of t-norms when the parameter is changed from zero to infinity. In this paper, we study a nonlinear optimization problem where the feasible region is formed as a system of fuzzy relational equations (FRE) defined by the Yager t-norm. We firstly investigate the resolution of the feasible region when it is defined with max-Yager composition and present some necessary and sufficient conditions for determining the feasibility and some

ARTICLE INFO

Article history:
Received 12, January 2018
Received in revised form 5, May 2018
Accepted 27 May 2018
Available online 01, June 2018

Keyword: Fuzzy relational equations; nonlinear optimization; genetic algorithm

AMS subject Classification: 05C78.

*Corresponding author: A. Ghodousian. Email: a.ghodousian@ut.ac.ir
†ajavan@ut.ac.ir
‡asieh.khoshnood@ut.ac.ir

1 Abstract continued:

procedures for simplifying the problem. Since the feasible solutions set of FREs is non-convex and the finding of all minimal solutions is an NP-hard problem, conventional nonlinear programming methods may involve high computation complexity. For these reasons, a method is used, which preserves the feasibility of new generated solutions. The proposed method does not need to initially find the minimal solutions. Also, it does not need to check the feasibility after generating the new solutions. Moreover, we present a technique to generate feasible max-Yager FREs as test problems for evaluating the performance of the current algorithm. The proposed method has been compared with Lu and Fangs algorithm. The obtained results confirm the high performance of the proposed method in solving such nonlinear problems.

2 Introduction

In this paper, we study the following nonlinear problem in which the constraints are formed as fuzzy relational equations defined by Yager t-norm:

$$\begin{aligned} \min \quad & f(x) \\ & A\varphi x = b \\ & x \in [0,1]^n \end{aligned} \tag{1}$$

where $I = \{1, 2 \ldots m\}$, $J = \{1, 2 \ldots n\}$, $A = (a_{ij})_{m \times n}, \quad 0 \le a_{ij} \le 1, (\forall i \in I \quad and \quad \forall j \in J)$ is a fuzzy matrix, $b = (b_i)_{m \times 1}, \ 0 \le b_i \le 1 \ (\forall j \in I)$ is a m-dimentional fuzzy vector, and φ is the max-Yager composition that is $\varphi(x,y) = T_Y^P(x,y) = max\{1 - [(1-x)^p + (1-y)^p]^{\frac{1}{p}}, 0\}$ in which $p > 0$.
if a_i is the i'th row of matrix A, then problem 1 can be expressed as follows:

$$\begin{aligned} \min \quad & f(x) \\ & \varphi(a_i, x) = b_i, \ i \in I \\ & x \in [0,1]^n \end{aligned} \tag{2}$$

where the constraints mean:

$$\begin{aligned} \varphi(a_i, x) &= \max_{j \in J} \{\varphi(a_{ij}, x_j)\} = \max_{j \in J} \{T_Y^P(a_{ij}, x_j)\} \\ &= \max_{j \in J} \{\max \{1 - [(1-a_{ij})^p + (1-x_j)^p]^{\frac{1}{p}}, 0\}\} \\ &= b_i \, , \qquad \forall i \in I \end{aligned} \tag{3}$$

As mentioned, the family $\{T_Y^P\}$ is strictly increasing in p. It can be easily shown that Yager t-norm $T_Y^P(x,y)$ converges to the basic fuzzy intersection $\min\{x,y\}$ as p goes to infinity and converges to Drastic product t-norm [8] as p approaches zero. Also, it is interesting to note that $T_Y^1(x, y) = \max\{x+y-1, 0\}$, that is, the Yager t-norm is converted to Lukasiewicz t-norm if $p = 1$. In [43] three feature types were presented based on the concept of information set for face recognition, which includes sigmoid and energy features, two features, viz. effective information set features-I and features-II and their combinations using t-norms and s-norms of Hamacher and Yager, and two hybrid features called Gabor-information set features and wavelet-information set features. In [46] the authors used Yager t-norm and t-conorm to investigate the performance of Fuzzy inference procedure of Fuzzy ID3 algorithm. In [47], the authors generalized a fixed-point theorem in fuzzy metric spaces by using a class of continuous t-norms known as ω-Yager t-norms, which was successfully used to prove the existence and uniqueness of solution for the recurrence equation associated with the probabilistic divide and conquer algorithms.

The problem to determine an unknown fuzzy relation R on universe of discourses $U \times V$ such that $A\varphi R = B$, where A and B are given fuzzy sets on U and V, respectively, and φ is an composite operation of fuzzy relations, is called the problem of fuzzy relational equations (FRE). Since Sanchez [54] proposed the resolution of FRE defined by max-min composition, different fuzzy relational equations were generalized in many theoretical aspects and utilized in many applied problems such as fuzzy control, discrete dynamic systems, prediction of fuzzy systems, fuzzy decision making, fuzzy pattern recognition, fuzzy clustering, image compression and reconstruction, fuzzy information retrieval, and so on [5,11,24,28,40,44,45,48,51,59,61,67]. For example, Klement et al. [31] presented the basic analytical and algebraic properties of triangular norms and important classes of fuzzy operators generalization such as Archimedean, strict and nilpotent t-norms. In [50] the author demonstrates how problems of interpolation and approximation of fuzzy functions are converted with solvability of systems of FRE. The authors in [45] used particular FRE for the compression/decompression of color images in the RGB and YUV spaces.

The solvability and the finding of solutions set are the primary (and the most fundamental) subject concerning FRE problems. Many studies have reported fuzzy relational equations with max-min and max-product compositions. Both compositions are special cases of the max-triangular-norm (max-t-norm). Di Nola et al. proved that the solution set of FRE (if it is nonempty) defined by continuous max-t-norm composition is often a non-convex set that is completely determined by one maximum solution and a finite number of minimal solutions [6]. This non-convexity property is one of two bottlenecks making major contribution to the increase in complexity of problems that are related to FRE, especially in the optimization problems subjected to a system of fuzzy relations. The other bottleneck is concerned with detecting the minimal solutions for FREs. Chen and Wang [2, 3] presented an algorithm for obtaining the logical representation of all minimal solutions and deduced that a polynomial-time algorithm to find all minimal solutions of FRE (with max-min compositions) may not exist. Also, Markovskii showed that solving max-product FRE is closely related to the covering problem which is an NP-hard problem [42]. In fact, the same result holds true for more general t-norms instead of the minimum and product operators [36, 37]. Lin et al. [37] demonstrated that all systems of max-continuous t-norm fuzzy relational equations, for example, max-product, max-continuous Archimedean t-norm and max-arithmetic mean are essentially equivalent, because they are all equivalent to the set covering problem. Over the last decades, the solvability of FRE defined with different max-t compositions has been investigated by many researches [49,52,53,55,57,58,62,66,70]. It is worth to mention that Li and Fang [35] provided a complete survey and a detailed discussion on fuzzy relational equations. They studied the relationship among generalized logical operators involved in the construction of FRE and introduced the classification of basic fuzzy relational equations.

Optimizing an objective function subjected to a system of fuzzy relational equations or inequalities (FRI) is one of the most interesting and on-going topics among the problems related to the FRE (or FRI) theory [1,9,13–27,33,38,56,63,68]. By far the most frequently studied aspect is the determination of a minimizer of a linear objective function and the use of the max-min composition [1,14]. So, it is an almost standard approach to translate this type of problem into a corresponding 0-1 integer linear programming problem, which is then solved using a branch and bound method [10,64]. In [32] an application of optimizing the linear objective with max-min composition was employed for the streaming media provider seeking a minimum cost while fulfilling the requirements assumed by a three-tier framework. Chang and Shieh [1] presented new theoretical results concerning the linear optimization problem constrained by fuzzy max-min relation equations by improving an upper bound on the optimal objective value. The topic of the linear optimization problem was also investigated with max-product operation [13, 26, 39]. Loetamonphong and Fang defined two sub-problems by separating negative and non-negative coefficients in the objective function and then obtained the optimal solution by combining those of the two sub-problems [39]. Also, in [26] and [13], some necessary conditions of the feasibility and simplification techniques were presented for solving FRE with max-product composition. Moreover, some studies have determined a more general operator of linear optimization with replacement of max-min and max-product compositions with a max-t-norm composition [18, 25, 33, 56], max-average composition [30, 63] or max-star composition [22].

Recently, many interesting generalizations of the linear and non-linear programming problems constrained by FRE or FRI have been introduced and developed based on composite operations and fuzzy relations used in the definition of the constraints, and some developments on the objective function of the problems [4, 7, 12, 14–17, 19, 20, 34, 38, 65]. For instance, the linear optimization of bipolar FRE was studied by some researchers where FRE was defined with max-min composition [12] and max-Lukasiewicz composition [34, 38]. In [34] the authors introduced the optimization problem subjected to a system of bipolar FRE defined as $X(A^+, A^-, b) = \{x \in [0,1]^m : x \circ A^+ \vee \tilde{x} \circ A^- = b\}$ where $\tilde{x_i} = 1 - x_i$ for each component of $\tilde{x} = (\tilde{x}_{i1\times m}$ and the notations "$\vee$" and "$\circ$" denote max operation and the max-Lukasiewicz composition, respectively. They translated the problem into a 0-1 integer linear programming problem which is then solved using well-developed techniques. In [38], the foregoing problem was solved by an analytical method based on the resolution and some structural properties of the feasible region (using a necessary condition for characterizing an optimal solution and a simplification process for reducing the problem). In [21], the authors focused on the algebraic structure of two fuzzy relational inequalities $A\varphi x \leq b^1$ and $D\varphi x \geq b^2$, and studied a mixed fuzzy system formed by the two preceding FRIs, where is an operator with (closed) convex solutions. Yang [69] studied the optimal solution of minimizing a linear objective function subject to fuzzy relational inequalities where the constraints defined as $a_{i1} \wedge x_1 + a_{i2} \wedge x_2 + \cdots + a_{in} \wedge x_n \geq b_i$ for $i = 1 \ldots m$ and $a \wedge b = \min\{a, b\}$. He presented an algorithm based on some properties of the minimal solutions of the FRI. Ghodousian et al. [17, 20] introduced FRI-FC problem $\min\{c^T x : A\varphi x \circ b\,,\ x \in [0,1]^n\}$, where φ is max-min composition and "$\circ$" denotes the relaxed or fuzzy version of the ordinary inequality "$\leq$".

Another interesting generalizations of such optimization problems are related to objective function. Wu et al. [65] represented an efficient method to optimize a linear fractional programming problem under FRE with max-Archimedean t-norm composition. Dempe and Ruziyeva [4] generalized the fuzzy linear optimization problem by considering fuzzy coefficients. Dubey et al. studied linear programming problems involving interval uncertainty modeled using intuitionistic fuzzy set [7]. If the objective function is $z(x) = \max_{i=1}^{n}\{\min\{c_i, x_i\}\}$ with $c_i \in [0,1]$, the model is called the latticized problem [60]. Also, Yang et al. [68] introduced another version of the latticized programming problem subject to max-prod fuzzy relation inequalities with application in the optimization management model of wireless communication emission base stations. The latticized problem was defined by minimizing objective function $z(x) = x_1 \vee x_2 \vee \cdots \vee x_n$ subject to feasible region $X(A, b) = \{x \in [0,1]^n : A \circ x \geq b\}$ where "$\circ$" denotes fuzzy max-product composition. They also presented an algorithm based on the resolution of the feasible region. On the other hand, Lu and Fang considered the single non-linear objective function and solved it with FRE constraints and max-min operator [41]. They proposed a genetic algorithm for solving the problem. Also, Ghodousian et al. [15, 16, 19] presented GA algorithms to solve the non-linear problem with FRE constraints defined by Lukasiewicz, Dubois Prade and Sugeno-Weber operators.

In this paper, we use the genetic algorithm proposed in [15] for solving problem (1), which keeps the search inside of the feasible region without finding any minimal solution and checking the feasibility of new generated solutions. For this purpose, the paper consists of three main parts. Firstly, we describe some structural details of FREs defined by the Yager t-norm such as the theoretical properties of the solutions set, necessary and sufficient conditions for the feasibility of the problem, some simplification processes and the existence of an especial convex subset of the feasible region. By utilizing the convex subset, the GA can easily generate a random feasible initial population. Then, an algorithm is presented based on the obtained theoretical properties. Finally, we provide some statistical and experimental results to evaluate the performance of the proposed algorithm. Since the feasibility of problem (1) is essentially dependent on the t-norm (Yager t-norm) used in the definition of the constraints, a method is also presented to construct feasible test problems. More precisely, we construct a feasible problem by randomly generating a fuzzy matrix A and a fuzzy vector b according to some criteria resulted from the necessary and sufficient conditions. It is proved that the max-Yager fuzzy relational equations constructed by this method is not empty. Moreover, a comparison is made between the current method and the method presented in [41].

The remainder of the paper is organized as follows. Section 2 takes a brief look at some basic results on the feasible solutions set of problem (1). In section 3, the GA algorithm is briefly described. A comparative study is presented in section 4 and, finally in section 5 the experimental results are demonstrated.

3 Some theoretical aspects of max-Yager FRE

3.1 Characterization of feasible solutions set

This section describes the basic definitions and structural properties concerning problem (1) that are used throughout the paper. For the sake of simplicity, let $S_{T_Y^P}(a_i, b_i)$ denote the feasible solutions set of i'th equation, that is $S_{T_Y^P}(a_i, b_i) = \{x \in [0,1]^n : \max_{j=1}^{n} \{T_Y^P(a_{ij}, x_j)\} = b_i\}$. Also, let $S_{T_Y^P}(A, b)$ denote the feasible solutions set of problem (1). Based on the foregoing notations, it is clear that
$S_{T_Y^P}(A, b) = \bigcap_{i \in I} S_{T_Y^P}(a_i, b_i)$.

Definition 3.1. For each $i \in I$, we define $J_i = \{j \in J : \ a_{ij} \geq b_i\}$.

According to definition 1, we have the following lemmas, which are easily proved by the monotonicity and identity law of t-norms, definition 1 and the definition of Yager t-norm.

Lemma 3.1. *For a fixed $i \in I$, $S_{T_Y^P}(a_i, b_i) \neq \varnothing$ if and only if $J_i \neq \varnothing$.*

Proof. The proof is similar to the proof of Lemma 3 in [15]. □

Definition 3.2. Suppose that $i \in I$ and $S_{T_Y^P}(a_i, b_i) \neq \varnothing$ (here, $J_i \neq \varnothing$ from lemma 3). Let $\widehat{x}_i = [(\widehat{x}_i)_1, (\widehat{x}_i)_2 \ldots (\widehat{x}_i)_n] \in [0,1]^n$ where the components are defined as follows:

$$(\widehat{x}_i)_k = \begin{cases} 1 - [(1-b_i)^p - (1-a_{ik})^p]^{\frac{1}{p}}, & k \in J_i \\ 1, & k \notin J_i \end{cases}, \forall k \in J$$

Also, for each $j \in J_i$, we define $\breve{x}_i = [(\breve{x}_i)_1, (\breve{x}_i)_2 \ldots (\breve{x}_i)_n] \in [0,1]^n$ such that:

$$\breve{x}_i(j)_k = \begin{cases} 1 - [(1-b_i)^p - (1-a_{ij})^p]^{\frac{1}{p}}, & b_i \neq 0 \text{ and } k = j \\ 0, & \text{otherwise} \end{cases}, \forall k \in J$$

The following theorem characterizes the feasible region of the i'th relational equation ($i \in I$).

Theorem 3.2. *Let $i \in I$. If $S_{T_Y^P}(a_i, b_i) \neq \varnothing$, then $S_{T_Y^P}(a_i, b_i) = \bigcup_{j \in J_i} [\breve{x}_i(j), \widehat{x}_i]$.*

Proof. For a more general case, see Corollary 2.3 in [21]. □

From theorem 1, $\widehat{x}_i$ is the unique maximum solution and $\breve{x}_i(j)$'s $(j \in J_i)$ are the minimal solutions of $S_{T_Y^P}(a_i, b_i)$.

Definition 3.3. Let $\widehat{x}_i$, $(i \in I)$ be the maximum solution of $S_{T_Y^P}(a_i, b_i)$. We define $\overline{X} = \min_{i \in I} \{\widehat{x}_i\}$.

Definition 3.4. Let $e : I \to J_i$ so that $e(i) = j \in J_i$, $\forall i \in I$, and let E be the set of all vectors e. For the sake of convenience, we represent each $e \in E$ as an m-dimensional vector $e = [j_1, j_2 \dots j_m]$ in which $j_k = e(k)$.

Definition 3.5. Let $e = [j_1, j_2 \dots j_m] \in E$. We define $X(e) = [\underline{X}(e)_1, \underline{X}(e)_2 \dots \underline{X}(e)_n] \in [0,1]^n$, where $\underline{X}(e)_j = \max_{i \in I} \{\breve{x}_i(e(i))_j\} = \max_{i \in I} \{\breve{x}_i(j_i)_j)\}, \forall j \in J$.

Theorem 2 below completely determines the feasible solutions set of problem (1).

Theorem 3.3. $S_{T_Y^P}(A, b) = \bigcup_{\epsilon \in E} [\underline{X}(e), \overline{X}]$.

Proof. Since $S_{T_Y^P}(A, b) = \bigcap_{i \in I} S_{T_Y^P}(a_i, b_i)$, from theorem 1 we have

$$
\begin{aligned}
S_{T_Y^P}(A, b) &= \bigcap_{i \in I} \bigcup_{j \in J_i} [\breve{x}_i(j), \widehat{x}_i] = \bigcap_{i \in I} \bigcup_{\epsilon \in E} [\breve{x}_i(e(i)), \widehat{x}_i] \\
&= \bigcup_{\epsilon \in E} \bigcap_{i \in I} [\breve{x}_i(e(i)), \widehat{x}_i] = \bigcup_{\epsilon \in E} [\max_{i \in I} \{\breve{x}_i(e(i))\}, \min_{i \in I} \{\widehat{x}_i\}] \\
&= \bigcup_{\epsilon \in E} [\underline{X}(e), \overline{X}]
\end{aligned}
$$

□

where the last equality is obtained by definitions 3 and 5.
As a consequence, it turns out that
overlineX is the unique maximum solution and $\underline{X}(e)$s $(einE)$ are the minimal solutions of $S_{T_Y^P}(A, b)$. Moreover, we have the following corollary that is directly resulted from theorem 2.

Corollary. ***first necessary and sufficient condition.*** *$S_{T_Y^P}(A, b) \neq \varnothing$ if and only if $\overline{X} \in S_{T_Y^P}(A, b)$.*

The following example illustrates the above-mentioned definitions.

Example 3.1. Consider the problem below with Yager t-norm

$$
\begin{bmatrix} 0.9 & 0.4 & 0.6 & 0.7 & 0.4 & 0.4 \\ 0.5 & 0.1 & 0.2 & 0.3 & 0.5 & 0.2 \\ 0.2 & 0.8 & 0.4 & 0.4 & 0.6 & 0.9 \\ 0.9 & 0.7 & 0.3 & 0.8 & 0.8 & 0.5 \\ 0.0 & 0.0 & 0.1 & 0.2 & 0.0 & 0.7 \end{bmatrix} \varphi x = \begin{bmatrix} 0.7 \\ 0.5 \\ 0.6 \\ 0.8 \\ 0.0 \end{bmatrix}
$$

where $\varphi(x,y) = T_Y^2(x,y) = \max\{1-[(1-x)^2+(1-y)^2]^{\frac{1}{2}},0\}$ (i.e., $p=2$). By definition 1, we have $J_1=\{1,4\}$, $J_2=\{1,5\}$, $J_3=\{2,5,6\}$, $J_4=\{1,4,5\}$ and $J_5=\{1,2,3,4,5,6\}$. The unique maximum solution and the minimal solutions of each equation are obtained by definition 2 as follows:

$\widehat{x}_1 = [0.7172,1,1,1,1,1], \widehat{x}_2 = [1,1,1,1,1,1], \widehat{x}_3 = [1,0.6536,1,1,1,0.6127],$
$\widehat{x}_4 = [0.8268,1,1,1,1,1], \widehat{x}_1 = [1,1,0.5641,0.4,1,0.0461].$
$\breve{x}_1(1) = [0.7172,0,0,0,0,0], \breve{x}_1(4) = [0,0,0,1,0,0],$
$\breve{x}_2(1) = [1,0,0,0,0,0], \breve{x}_2(5) = [0,0,0,0,1,0],$
$\breve{x}_3(2) = [0,0.6536,0,0,0,0], \breve{x}_3(5) = [0,0,0,0,1,0], \breve{x}_3(6) = [0,0,0,0,0,0.6127]$
$\breve{x}_4(1) = [0.8268,0,0,0,0,0], \breve{x}_4(4) = [0,0,0,1,0,0], \breve{x}_4(5) = [0,0,0,0,1,0],$
$\breve{x}_5(j) = [0,0,0,0,0,0], \ j \in \{1,2,3,4,5,6\}$

Therefore, by theorem 1 we have $S_{T_Y^P}(a_1,b_1) = [\breve{x}_1(1),\widehat{x}_1] \cup [\breve{x}_1(4),\widehat{x}_1]$, $S_{T_Y^P}(a_2,b_2) = [\breve{x}_2(1),\widehat{x}_2] \cup [\breve{x}_2(5),\widehat{x}_2]$, $S_{T_Y^P}(a_3,b_3) = [\breve{x}_3(2),\widehat{x}_3] \cup [\breve{x}_3(5),\widehat{x}_3] \cup [\breve{x}_3(6),\widehat{x}_3]$, $S_{T_Y^P}(a_4,b_4) = [\breve{x}_4(1),\widehat{x}_4] \cup [\breve{x}_4(4),\widehat{x}_4] \cup [\breve{x}_4(5),\widehat{x}_4]$, and $S_{T_Y^P}(a_5,b_5) = [\mathbf{0}_{1\times 6},\widehat{x}_5]$ where $\mathbf{0}_{1\times 6}$ is a zero vector. From definition 3, $\overline{X} = [0.7172,0.6536,0.5641,0.4,1,0.0461]$. It is easy to verify that $\overline{X} = S_{T_Y^P}(A,b)$. Therefore, the above problem is feasible by corollary 1. Finally, the cardinality of set E is equal to 36 (definition 4). So, we have 36 solutions $\underline{X}(e)$ associated to 36 vectors e. For example, for $e=[1,5,2,5,5]$, we obtain $\underline{X}(e) = \max\{\breve{x}_1(1),\breve{x}_2(5),\breve{x}_3(2),\breve{x}_4(5),\breve{x}_5(5)\}$ from definition 5 that means
$\underline{X}(e) = [0.7172,0.6536,0,0,1,0]$.

3.2 Simplification processes

In practice, there are often some components of matrix A that have no effect on the solutions to problem (1). Therefore, we can simplif11y the problem by changing the values of these components to zeros. For this reason, various simplification processes have been proposed by researchers. We refer the interesting reader to [21] where a brief review of such these processes is given. Here, we present two simplification techniques based on the Yager t-norm.

Definition 3.6. If a value changing in an element, say a_{ij}, of a given fuzzy relation matrix A has no effect on the solutions of problem (1), this value changing is said to be an equivalence operation.

Corollary. *Suppose that* $T_Y^P(a_{ij_0},x_{j_0})$, $\forall x \in S_{T_Y^P}(A,b)$. *In this case, it is obvious that* $\max_{j=1}^{n}\{T_Y^P(a_{ij},x_j)\} = b_i$ *is equivalent to* $\max_{j=1,\, j\neq j_0}^{n}\{T_Y^P(a_{ij},x_j)\} = b_i$*, that is, "resetting* a_{ij_0} *to zero" has no effect on the solutions of problem (1) (since component* a_{ij_0} *only appears in the i'th constraint of problem (1)). Therefore, if* $T_Y^P(a_{ij_0},x_{j_0}) < b_i$, $\forall x \in S_{T_Y^P}(A,b)$ *, then "resetting* a_{ij_0} *to zero" is an equivalence operation.*

Lemma 3.4. ***(first simplification)***. *Suppose that $j_0 \in J_i$, for some $i \in I$ and $j_0 \in J$. Then, "resetting a_{ij_0} to zero" is an equivalence operation.*

Proof. From corollary 2, it is sufficient to show that $T_Y^P(a_{ij_0}, x_{j_0}) < b_i$, $\forall x \in S_{T_Y^P}(A, b)$. But, from lemma 1 we have $T_Y^P(a_{ij_0}, x_{j_0}) < b_i$, $\forall x_{j_0} \in [0, 1]$. Thus, $T_Y^P(a_{ij_0}, x_{j_0}) < b_i$, $\forall x \in S_{T_Y^P}(A, b)$. □

Lemma 3.5. ***(second simplification)***. *Suppose that $j_0 \in J_{j_1}$ and $b_{i_1} \neq 0$, where $i_1 \in I$ and $j_0 \in J$. If $j_0 \in J_{i_2}$ for some $i_2 \in I (i_1 \neq i_2)$ and $[(1-b_{i_2})^p - (1-a_{i_2j_0})^p]^{\frac{1}{p}} > [(1-b_{i_1})^p - (1-a_{i_1j_0})^p]^{\frac{1}{p}}$, then "resetting $a_{i_1j_0}$ to zero" is an equivalence operation.*

Proof. Similar to the proof of lemma 4, we show that $T_Y^P(a_{i_1j_0}, x_{j_0}) < b_i$, $\forall x \in S_{T_Y^P}(A, b)$. Consider an arbitrary feasible solution $x \in S_{T_Y^P}(A, b)$. Since $x \in S_{T_Y^P}(A, b)$, it turns out that $T_Y^P(a_{i_1j_0}, x_{j_0}) > b_{i_1}$ never holds. So, assume that $T_Y^P(a_{i_1j_0}, x_{j_0}) = b_{i_1}$, that is, $\max\{1 - [(1-a_{i_1j_0})^p + (1-x_{j_0})^p]^{\frac{1}{p},0]}\} = b_{i_1}$. Since $b_{i_1} \neq 0$, we conclude that $1 - [(1-a_{i_1j_0})^p + (1-x_{j_0})^p]^{\frac{1}{p}} = b_{i_1}$, or equivalently $x_{j_0} = 1 - [(1-a_{i_1j_0})^p + (1-x_{j_0})^p]^{\frac{1}{p}} = b_{i_1}$. Now, from $[(1-b_{i_2})^p - (1-a_{i_2j_0})^p]^{\frac{1}{p}} > [(1-b_{i_1})^p - (1-a_{i_1j_0})^p]^{\frac{1}{p}}$, we obtain $x_{j_0} > [(1-b_{i_1})^p - (1-a_{i_1j_0})^p]^{\frac{1}{p}}$. Therefore, from lemma 2 (part (a)), we have $T_Y^P(a_{i_2j_0}, x_{j_0}) > b_{i_2}$ that contradicts $x \in S_{T_Y^P}(A, b)$. □

We give an example to illustrate the above two simplification processes.

Example 3.2. Consider the problem presented in example 1. From the first simplification (lemma 4), "resetting the following components a_{ij} to zeros" are equivalence operations: $a_{12}, a_{13}, a_{15}, a_{16}; a_{22}, a_{23}, a_{24}, a_{26}; a_{31}, a_{33}, a_{34}; a_{42}, a_{43}, a_{46}$; in all of these cases, $a_{ij} < b_i$, that is, $j \notin J_i$. Moreover, from the second simplification (lemma 5), we can change the values of components $a_{14}, a_{21}, a_{36}, a_{41}$, and a_{44} to zeros with no effect on the solutions set of the problem. For example, since $a_{41} > b_4$(i.e. $1 \in J_4$), $b_4 \neq 0$, $a_{11} > b_1$ (i.e. $1 \in J_1$) and
$0.2828 = [(1-b_1)^p - (1-a_{11})^p]^{\frac{1}{p}} > [(1-b_4)^p - (1-a_{41})^p]^{\frac{1}{p}} = 0.1732$
"resetting a_{41} to zero" is an equivalence operation.

In addition to simplifying the problem, a necessary and sufficient condition is also derived from lemma 5. Before formally presenting the condition, some useful notations are introduced. Let $\tilde{A}$ denote the simplified matrix resulted from A after applying the simplification processes (lemmas 4 and 5). Also, similar to definition 1, assume that $\tilde{J}_i = \{j \in J : \tilde{a}_{ij} \geq b_i \ (i \in I)$ where $\tilde{a}_{ij}$ denotes (i, j)'th component of matrix $\tilde{A}$. The following theorem gives a necessary and sufficient condition for the feasibility of problem (1).

Theorem 3.6. *(second necessary and sufficient condition).$S_{T_Y^P} \neq \varnothing$ if and only if $\tilde{J}_i \neq \varnothing, \forall i \in I$.*

Proof. Since $S_{T_Y^P}(A, b) = S_{T_Y^P}(\tilde{A}, b)$ from lemmas 4 and 5, it is sufficient to show that $S_{T_Y^P}(\tilde{A}, b) \neq \varnothing$ if and only if $\tilde{J}_i \neq \varnothing$, $\forall i \in I$. Let $S_{T_Y^P}(\tilde{A}, b) \neq \varnothing$. Therefore, $S_{T_Y^P}(\tilde{a}_i, b_i) \neq$

φ, $\forall i \in I$, where $\tilde{a}_i$ denotes i'th row of matrix $\tilde{A}$. Now, lemma 3 implies $\tilde{J}_i \neq \varphi$, $\forall i \in I$. Conversely, suppose that $\tilde{J}_i \neq \varphi$, $\forall i \in I$. Again, by using lemma 3 we have $\tilde{J}_i \neq \varphi$, $\forall i \in I$. By contradiction, suppose that $S_{T_Y^P}(\tilde{A}, b) = \varphi$. Therefore, $\overline{X} \notin S_{T_Y^P}(\tilde{A}, b)$ from corollary 1, and then there exists $i_0 \in I$ such that $\overline{X} \notin S_{T_Y^P}(\tilde{a}_{i_0}, b_{i_0})$. Since $\max_{j \notin J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} < b_{i_0}$(from lemma 1), we must have either $\max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} > b_{i_0}$ or $\max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} < b_{i_0}$. Anyway, since $\overline{X} \leq \widehat{x}_{i_0}$ (i.e. $\overline{X}_j \leq (\widehat{x}_{i_0})_j$, $\forall j \in J$), we have $\max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} \leq \max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, (\widehat{x}_{i_0})_j)\} = b_{i_0}$, and then the former case (i.e. $\max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} > b_{i_0}$) never holds. Therefore, $\max_{j \in J_i}\{T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j)\} < b_{i_0}$ that implies $b_{i_0} \neq 0$ and $T_Y^P(\tilde{a}_{i_0j}, \overline{X}_j) < b_{i_0}$, $\forall j \in \tilde{J}_{i_0}$. Hence, by lemma 2, we must have $\overline{X}_j < 1 - [(1-b_{i_0})^p - (1-\tilde{a}_{i_0j})^p]^{\frac{1}{p}}$, $\forall j \in \tilde{J}_{i_0}$. On the other hand, $[(1-b_{i_0})^p - (1-\tilde{a}_{i_0j})^p]^{\frac{1}{p}} \geq 0$, $\forall j \in \tilde{J}_{i_0}$. Therefore, $\overline{X}_j < 1$, $\forall j \in \tilde{J}_{i_0}$, and then from definitions 2 and 3, for each $\forall j \in \tilde{J}_{i_0}$ there must exists $i_j \in I$ such that $j \in \tilde{J}_{i_j}$ and $\overline{X}_j = (\widehat{x}_{i_j})_j = 1 - [(1-b_{i_j})^p - (1-\tilde{a}_{i_jj})^p]^{\frac{1}{p}}$. Until now, we proved that $b_{i_0} \neq 0$ and for each $j \in \tilde{J}_{i_0}$, there exist $i_j \in I$ such that $j \in \tilde{J}_{i_j}$ and $[(1-b_{i_j})^p - (1-\tilde{a}_{i_jj})^p]^{\frac{1}{p}} > [(1-b_{i_0})^p - (1-\tilde{a}_{i_0j})^p]^{\frac{1}{p}}$ (because, $1 - [(1-b_{i_j})^p - (1-\tilde{a}_{i_jj})^p]^{\frac{1}{p}} = \overline{X}_j < 1 - [(1-b_{i_0})^p - (1-\tilde{a}_{i_0j})^p]^{\frac{1}{p}}$). But in these cases, we must have $\tilde{a}_{i_0j} = 0$ ($\forall j \in \tilde{J}_{i_0}$) from the second simplification process. Therefore, $\tilde{a}_{i_0j} < b_{i_0}$($\forall j \in \tilde{J}_{i_0}$) that is a contradiction. □

Remark. Since $S_{T_Y^P}(A, b) = S_{T_Y^P}(\tilde{A}, b)$(from lemmas 4 and 5), we can rewrite all the previous definitions and results in a simpler manner by replacing $\tilde{J}_i$ with $J_i (i \in I)$.

4 The proposed GA for solving problem (1)

In this section, the genetic algorithm proposed in [15] is briefly discussed. Since the feasible region of problem (1) is non-convex, a convex subset of the feasible region is firstly introduced. Consequently, the proposed GA can easily generate the initial population by randomly choosing individuals from this convex feasible subset. At the last part of this section, a method is presented to generate random feasible max-Yager fuzzy relational equations.

4.1 Initialization

The initial population is given by randomly generating the individuals inside the feasible region. For this purpose, we firstly find a convex subset of the feasible solutions set, that is, we find set F such that $F \subseteq S_{T_Y^P}(A, b)$ and F is convex. Then, the initial population is generated by randomly selecting individuals from set F.

Definition 4.1. Suppose that $S_{T_Y^P}(\tilde{A}, b) \neq \varnothing$. For each $i \in I$, let $\breve{x}_i = [(\breve{x}_i)_1, (\breve{x}_i)_2 \dots (\breve{x}_i)_n] \in [0, 1]^n$ where the components are defined as follows:

$$(\breve{x}_i)_k = \begin{cases} 1 - [(1-b_i)^p - (1-a_{ik})^p]^{\frac{1}{p}}, & b_i \neq 0 \; and \; kin\tilde{J}_i \\ 0, & otherwise \end{cases}, \forall k \in J$$

Also, we define $\underline{X} = \max_{i \in I} \{\tilde{x}_i\}$.

Remark. According to definition 2 and remark 1, it is clear that for a fixed $i \in I$ and $j \in \tilde{J}_i$, $\breve{x}_i(j)_k \leq (\breve{x}_i)_k$ ($\forall k \in J$). Therefore, from definitions 5 and 7 we have $\underline{X}(e)_k = \max_{i \in I} \{\tilde{x}_i(e(i))_k\} = \max_{i \in I} \{\tilde{x}_i(j_i)_k\} \leq \max_{i \in I} \{(\tilde{x}_i)_k\} = \underline{X}_k$, $\forall kinJ$ and $\forall einE$. Thus, $\underline{X}(e) \leq \underline{X}$, $\forall einE$. Now, Suppose that $S_{T_Y^P}(\tilde{A}, b) \neq \varnothing$ and $F = \{x \in [0,1]^n : \underline{X} \leq x \leq \overline{X}\}$. Then, $F \subseteq S_{T_Y^P}(\tilde{A}, b)$ and is a convex set [15].

Example 4.1. Consider the problem presented in example 1, where
$\overline{X} = [0.7172, 0.6536, 0.5641, 0.4, 1, 0.0461]$. Also, according to example 2, the simplified matrix $\tilde{A}$ is

$$\tilde{A} = \begin{bmatrix} 0.9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0.8 & 0 & 0 & 0.6 & 0 \\ 0 & 0 & 0 & 0 & 0.8 & 0 \\ 0 & 0 & 0.1 & 0.2 & 0 & 0.7 \end{bmatrix}$$

From definition 7, we have $\tilde{x}_1 = [0.7172, 0, 0, 0, 0, 0]$, $\tilde{x}_2 = [0, 0, 0, 0, 1, 0]$,
$\tilde{x}_3 = [0, 0.6536, 0, 0, 1, 0]$, $\tilde{x}_4 = [0, 0, 0, 0, 1, 0]$, $\tilde{x}_5 = [0, 0, 0, 0, 0, 0]$,
and then $\underline{X} = \max_{i=1}^{5} \{\tilde{x}_i\} = [0.7172, 0.6536, 0, 0, 1, 0]$. Therefore, set $F = [\underline{X}, \overline{X}]$ is obtained as a collection of intervals:
$F = [\underline{X}, \overline{X}] = [0.7172, 0.6536, [0, 0.5641], [0, 0.4], 1, [0, 0.0461]]$
By generating random numbers in the corresponding intervals, we acquire one initial individual: $x = [0.7172, 0.6536, 0.4298, 0.3, 1, 0.0211]$.

The algorithm for generating the initial population is simply obtained as follows:

Algorithm 1 Initial Population

Get fuzzy matrix A, fuzzy vector b and population size S_{pop}
If $\overline{X} \notin S_{T_Y^P}(A, b)$, then stop; the problem is unfeasible (corollary 1).
For $i = 1 \ldots S_{pop}$
  Generate a random n-dimensional solution $pop(i)$ in the interval $[\underline{X}, \overline{X}]$
End

4.2 Selection strategy

Suppose that the individuals in the population are sorted according to their ranks from the best to worst, that is, individual $pop(r)$ has rank r. The probability P_r of choosing the r'th individual is given by the following formulas:

$P_r = \frac{W_r}{\sum_{k=1}^{S_{pop}} W_k}$, $W_r = \frac{1}{\sqrt{2\pi} q S_{pop}} e^{-\frac{1}{2}[\frac{r-1}{qS_{pop}}]^2}$
where the weight to be a value of the Gaussian function with argument r, mean 1, and standard deviation qS_{pop}, where q is a parameter of the algorithm.

4.3 Mutation operator

As usual, suppose that $S_{T_Y^P}(A, b) \neq \varnothing$. So, from theorem 3 we have $\tilde{J}_i \neq \varnothing$, $\forall i \in I$, where $\tilde{J}_i = \{j \in J : \tilde{a}_{ij} \geq b_i\}$, $\forall i \in I$ (see definition 1 and remark 1).

Definition 4.2. Let $I^+ = \{i \in I : b_i \neq 0\}$. So, we define $D = \{j \in J : if \exists i \in I^+ such that j \in \tilde{J}_i \Rightarrow |\tilde{J}_i| > 1\}$, where $|\tilde{J}_i|$ denotes the cardinality of set $\tilde{J}_i$.

The mutation operator is defined as follows:

Algorithm 2 Mutation Operator

Get the matrix $\tilde{A}$, vector b and a selected solution $\dot{x} = [\dot{x}_1 \dots \dot{x}_n]$
While $D \neq \varnothing$
  Set $x' \leftarrow x$
  Randomly choose $j_0 \in D$, and set $x'_{j_0} = 0$
  If x' is feasible, goto Crossover operator, otherwise set $D = D - \{j_0\}$

4.4 Crossover operator

In section 2, it was proved that $\overline{X}$ is the unique maximum solution of $S_{T_Y^P}(A, b)$. By using this result, the crossover operator is stated as follows:

Algorithm 3 Crossover Operator

Get the maximum solution $\overline{X}$, the new solution x' (generated by Alg. 2),
  and one parent $pop(k)$ (for some $k = 1 \dots S_{pop}$)
Generate a random number $\lambda_1 \in [0, 1]$. Set $x_{new1} = \lambda_1 x' + (1 - \lambda_1)\overline{X}$
Let $\lambda_2 = \min_{j=1, j \neq k}^{S_{pop}} \|pop(k) - pop(j)\|$, and $d = \overline{X} - pop(k)$ Set $x_{new2} = pop(k) + \min\{\lambda_2, 1\} d$

4.5 Construction of test problems

There are usually several ways to generate a feasible FRE defined with different t-norms. In what follows, we present a procedure to generate random feasible max-Yager fuzzy relational equations:
From step 4 of the above algorithm, we note that if $\theta \leq 0.5$, then we will have $a_{kj_i} \in [0, b_k)$, and therefore $j_i \notin J_k$. Also, if $\theta > 0.5$ and $(1-b_k)^p < (1-b_i)^p - (1-a_{ij_i})^p$, then $a_{kj_i} \in [b_k, 1]$

Algorithm 4 Construction of Feasible Max-Yager FRE

Randomly select m columns $\{j_1 \dots j_m\}$ from $J = \{1 \dots n\}$
Generate vector b whose elements are random numbers from $[0, 1]$
For $i \in \{1 \dots m\}$
  Assign a random number from $[b_i, 1]$ to a_{ij_i}
End
For $i \in \{1 \dots m\}$
  If $b_i \neq 0$
    For each $k \in \{1 \dots m\} - \{i\}$, generate a random number θ from $[0, 1]$
      If $\theta \leq 0.5$, assign a random number from $[0, b_k)$ to a_{kj_i}
      Else If $(1 - b_k)^p < (1 - b_i)^p - (1 - a_{ij_i})^p$, assign a random number from $[b_k, 1]$ to a_{kj_i}
      Else assign a random number from $[0, 1-[(1-a_{ij_i})^p-(1-b_i)^p+(1-b_k)^p]^{\frac{1}{p}}]$ to a_{kj_i}
End
For each $i \in \{1 \dots m\}$ and each $j \notin \{j_1 \dots j_m\}$
  Assign a random number from $[0, 1]$ to a_{ij}
End

. In this case, after applying the algorithm we will have $[(1 - b_k)^p - (1 - a_{kj_i})^p]^{\frac{1}{p}} \leq [(1 - b_i)^p - (1 - a_{ij_i})^p]^{\frac{1}{p}}$. By the following theorem, it is proved that algorithm 4 always generates random feasible max-Yager fuzzy relational equations.

Theorem 4.1. *The solutions set $S_{T_Y^P}(A, b)$ of FRE (with Yager t-norm) constructed by algorithm 4 is not empty.*

Proof. According to step 3 of the algorithm, $j_i \in J_i$, $\forall i \in I$. Therefore, $J_i \neq \varnothing$, $\forall i \in I$. To complete the proof, we show that $j_i \in \tilde{J}_i$, $\forall i \in I$. By contradiction, suppose that the second simplification process reset a_{ij_i} to zero, for some $i \in I$. Hence, $b_i \neq 0$ and there must exists some $k \in I$ $(k \neq i)$ such that $[(1-b_k)^p-(1-a_{kj_i})^p]^{\frac{1}{p}} > [(1-b_i)^p-(1-a_{ij_i})^p]^{\frac{1}{p}}$ and $j_i \in J_k$. But in this case, we must have $(1 - b_k)^p > (1 - b_i)^p - (1 - a_{ij_i})^p$, and then $a_{kj_i} > 1 - [(1 - aij_i)^p - (1 - b_i)^p + (1 - b_k)^p]^{\frac{1}{p}}$, that contradicts step 4. □

5 Experimental Results

In this section, we present the experimental results for evaluating the performance of the proposed algorithm. Firstly, we apply the current algorithm to 8 test problems described in Appendix A. The test problems have been randomly generated in different sizes by algorithm 4 given in section 3. Since the objective function is an ordinary nonlinear function, we take some objective functions from the well-known source: Test Examples for Nonlinear Programming Codes [29]. In section 5.2, we make a comparison against the related GA proposed in [41]. To perform a fair comparison, we follow the same experimental setup for the parameters $\theta = 0.5$, $\xi = 0.01$, $\lambda = 0.995$ and $\gamma = 1.005$ as suggested by the authors in [41]. Since the authors did not explicitly reported the size of the population, we consider $S_{pop} = 50$ for all methods. As mentioned before, we set $q = 0.1$ in relation (2) for the current GA. Moreover, in order to compare the proposed algorithm with max-min GA [41], we modified all the definitions used in the current method based on the minimum t-norm. For example, we used the simplification process presented in [41]. Finally, 30 experiments are performed for all the methods and for eight test problems reported in Appendix B, that is, each of the methods is executed 30 times for each test problem. All the test problems included in Appendix A, have been defined by considering $p = 2$ in T_Y^P. Also, the maximum number of iterations is equal to 100 for all the methods.

5.1 Performance of the max-Yager GA

To verify the solutions found by the current method, the optimal solutions of the test problems are also needed. Since $S_{T_Y^P}(A, b)$ is formed as the union of the finite number of convex closed cells (theorem 2), the optimal solutions are also acquired by the following procedure: 1. Computing all the convex cells of the Yager FRE. 2. Searching the optimal solution for each convex cell. 3. Finding the global optimum by comparing these local optimal solutions.
The computational results of the eight test problems are shown in Table 1 and Figures 1-8. In Table 1, the results are averaged over 30 runs and the average best-so-far solution, average mean fitness function and median of the best solution in the last iteration are reported. Table 2 includes the best results found by the current algorithm and the above procedure. According to Table 2, the optimal solutions computed by the algorithm and the optimal solutions obtained by the above procedure match very well. Tables 1 and 2, demonstrate the attractive ability of the algorithm to detect the optimal solutions of problem (1). Also, the good convergence rate of the algorithm could be concluded from Table 1 and figures 1-8.

Table 1: Results of applying the max-Yager GA to the eight test problems. The results have been averaged over 30 runs. Maximum number of iterations=100.

Test problems	Average best-so-far	Median best-so-far	Average mean fitness
A.1	10.918379	10.918379	10.919222
A.2	-0.461955	-0.461955	-0.461904
A.3	-0.939706	-0.939706	-0.938272
A.4	2.621036	2.621036	2.623218
A.5	33.489036	33.489038	33.494282
A.6	-0.302549	-0.302549	-0.302529
A.7	-0.788851	-0.788851	-0.788426
A.8	33.28396	33.28396	33.28399

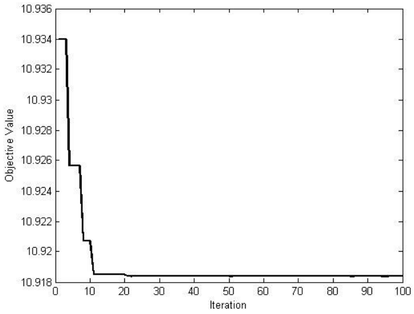

Figure 1: The performance of the proposed algorithm on test problem 1.

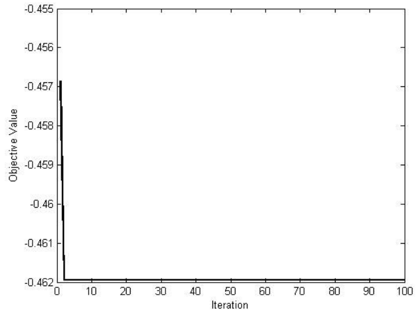

Figure 2: The performance of the proposed algorithm on test problem 2.

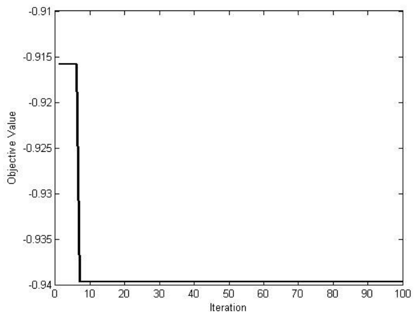

Figure 3: The performance of the max-Yager GA on test problem 3.

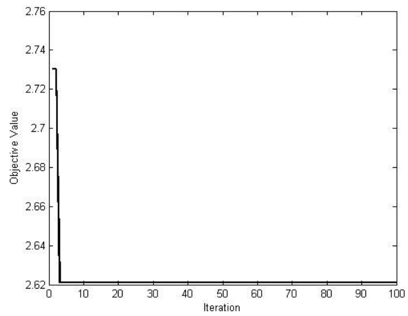

Figure 4: The performance of the max-Yager GA on test problem 2.

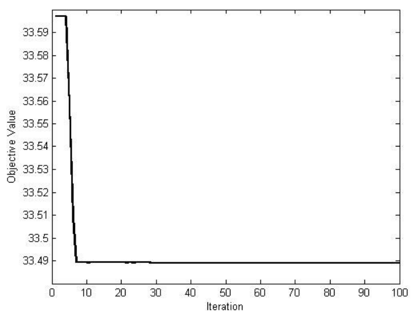

Figure 5: The performance of the proposed algorithm on test problem 5.

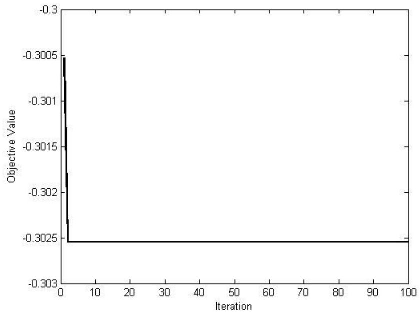

Figure 6: The performance of the proposed algorithm on test problem 2.

Table 2: Comparison of the solutions found by the current method and the optimal values of the test problems.

Test problems	Solutions of max-Yager GA	Optimal values
A.1	10.918379	10.918379
A.2	-0.461955	-0.46192
A.3	-0.939706	-0.93971
A.4	2.621036	2.621031
A.5	33.489036	33.4861
A.6	-0.302549	-0.302549
A.7	-0.788851	-0.788851
A.8	33.28396	33.2835

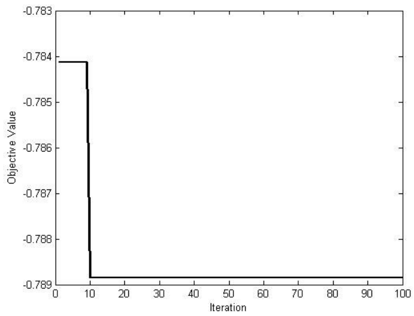

Figure 7: The performance of the proposed algorithm on test problem 7.

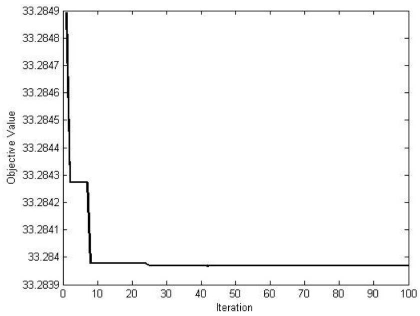

Figure 8: The performance of the proposed algorithm on test problem 8.

5.2 Comparisons with other works

As mentioned before, we can make a comparison between the current algorithm and max-min GA [41]. We apply the current algorithm (modified for the minimum t-norm) to the test problems by considering φ as the minimum t-norm. The results are shown in Table 3 including the optimal objective values found by the current method and max-min GA. As is shown in this table, the current method finds better solutions for test problems 1, 5 and 6, and the same solutions for the other test problems. Table 4 shows that the current algorithm finds the optimal values faster than max-min GA and hence has a higher convergence rate, even for the same solutions. The only exception is test problem 8 in which all the results are the same. In all the cases, results marked with "*" indicate the better cases.

Conclusion

In this paper, we investigated the resolution of FRE defined by the Yager t-norm and introduced a nonlinear problem with the max-Yager fuzzy relational equations. Two nec-

Table 3: Best results found by the current algorithm and Lu and Fangs method.

Test problems	Lu and Fang	Current algorithm
B.1	8.4296755	8.4296754*
B.2	-1.3888	-1.3888
B.3	0	0
B.4	5.0909	5.0909
B.5	71.1011	71.0968*
B.6	-0.3291	-0.4175
B.7	-0.6737	-0.6737*
B.8	93.9796	93.9796

essary and sufficient conditions were derived to determine the feasibility of the problem. In order to simplify the problem, we presented two simplification approaches depending on the Yager t-norm. A genetic algorithm was used for solving the nonlinear optimization problems constrained by the max-Yager FRE. Moreover, we presented a method for generating feasible max-Yager FREs as test problems for the performance evaluation of the proposed algorithm. Experiments were performed with the proposed method in the generated feasible test problems. We conclude that the proposed method can find the optimal solutions for all the cases with a great convergence rate. Moreover, a comparison was made between the proposed method and Lu and Fangs method, which solve the nonlinear optimization problems subjected to the FREs defined by max-min composition. The results showed that the proposed method finds better solutions compared with the solutions obtained by Lu and Fangs algorithm. As future works, we aim at testing current algorithm in other type of nonlinear optimization problems whose constraints are defined as FRE or FRI with other well-known t-norms.

Table 4: A Comparison between the results found by the current algorithm and Lu and Fangs algorithm.

Test problems		Lu and Fang	Current algorithm
B.1	Average best-so-far	8.4296755	8.4296796*
	Median best-so-far	8.4296755	8.4296755
	Average mean fitness	8.4296755	8.4398745*
B.2	Average best-so-far	-1.3888	-1.3888
	Median best-so-far	-1.3888	-1.3888
	Average mean fitness	-1.3877	-1.3886*
B.3	Average best-so-far	0	0
	Median best-so-far	0	0
	Average mean fitness	7.1462e-07	0*
B.4	Average best-so-far	5.0909	5.0909
	Median best-so-far	5.0909	5.0909
	Average mean fitness	5.0910	5.0908*
B.5	Average best-so-far	71.1011	71.0969*
	Median best-so-far	71.1011	71.0968*
	Average mean fitness	71.1327	71.1216*
B.6	Average best-so-far	-0.3291	-0.4175*
	Median best-so-far	-0.3291	-0.4175*
	Average mean fitness	-0.3287	-0.4162*
B.7	Average best-so-far	-0.6737	-0.6737
	Median best-so-far	-0.6737	-0.6737
	Average mean fitness	-0.6736	-0.6737*
B.8	Average best-so-far	93.9796	93.9796
	Median best-so-far	93.9796	93.9796
	Average mean fitness	93.9796	93.9796

Appendix A

Test Problem A.1:

$f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$

$b^T = [0.2077, 0.4709, 0.8443]$

$$A = \begin{bmatrix} 0.4302 & 0.4464 & 0.0741 & 0.0751 \\ 0.1848 & 0.1603 & 0.4628 & 0.5929 \\ 0.9049 & 0.1707 & 0.8746 & 0.4210 \end{bmatrix}$$

Test Problem A.2:

$f(x) = x_1 - x_2 - x_3 - x_1x_3 + x_1x_4 + x_2x_3 - x_2x_4 + x_4x_5$

$b^T = [0.0871, 0.3713, 0.2455, 0.1801]$

$$\begin{bmatrix} 0.5801 & 0.7557 & 0.0705 & 0.0568 & 0.0612 \\ 0.6871 & 0.3217 & 0.6975 & 0.6199 & 0.8560 \\ 0.0363 & 0.5551 & 0.1511 & 0.8654 & 0.1547 \\ 0.8343 & 0.0525 & 0.1708 & 0.0591 & 0.5315 \end{bmatrix}$$

Test Problem A.3:

$f(x) = x_1 - x_2 - ln(1 + x_3x_4x_5) - x_6$

$b^T = [0.5531, 0.2219, 0.9524, 0.8888]$

$$\begin{bmatrix} 0.4432 & 0.4430 & 0.0774 & 0.7655 & 0.2581 & 0.2074 \\ 0.2629 & 0.0780 & 0.8817 & 0.2126 & 0.5757 & 0.1161 \\ 0.9554 & 0.9857 & 0.5055 & 0.5656 & 0.1704 & 0.1256 \\ 0.2025 & 0.1048 & 0.5135 & 0.7883 & 0.8020 & 0.9774 \end{bmatrix}$$

Test Problem A.4:

$f(x) = x_1 + 2x_2 + 4x_5 + e^{x_1x_4 - x_6}$

$b^T = [0.9296, 0.6019, 0.1510, 0.5746, 0.0953]$

$$\begin{bmatrix} 0.9486 & 0.9505 & 0.1827 & 0.5498 & 0.2400 & 0.0183 \\ 0.0787 & 0.5247 & 0.8651 & 0.4489 & 0.6457 & 0.1841 \\ 0.9116 & 0.1440 & 0.0188 & 0.0400 & 0.1404 & 0.0818 \\ 0.8890 & 0.3096 & 0.4366 & 0.8204 & 0.5103 & 0.3569 \\ 0.0567 & 0.0415 & 0.0748 & 0.0888 & 0.0672 & 0.1593 \end{bmatrix}$$

Test Problem A.5:

$f(x) = \sum_{k=1}^{6} [100(x_{k+1} - x_k^2)^2 + (1 - x_k)^2]$

$b^T = [0.3205, 0.3143, 0.4007, 0.7064, 0.3223]$

$$\begin{bmatrix} 0.1588 & 0.7224 & 0.1207 & 0.6127 & 0.8826 & 0.7250 & 0.6135 \\ 0.2848 & 0.9535 & 0.2324 & 0.3272 & 0.9862 & 0.0513 & 0.1372 \\ 0.0671 & 0.2631 & 0.7020 & 0.9393 & 0.5195 & 0.5549 & 0.5847 \\ 0.9175 & 0.4171 & 0.7517 & 0.0661 & 0.1765 & 0.7090 & 0.3740 \\ 0.1342 & 0.8735 & 0.2747 & 0.8126 & 0.3290 & 0.1664 & 0.9600 \end{bmatrix}$$

Test Problem A.6:

$f(x) = -0.5(x_1x_4 - x_2x_3 + x_2x_6 - x_5x_6 + x_5x_4 - x_6x_7)$

$b^T = [0.2793, 0.9500, 0.7740, 0.8420, 0.4028, 0.9032]$

$$\begin{bmatrix} 0.1285 & 0.0451 & 0.1427 & 0.1872 & 0.2295 & 0.7203 & 0.6670 \\ 0.9561 & 0.9761 & 0.2078 & 0.9746 & 0.9781 & 0.4570 & 0.4021 \\ 0.6666 & 0.8612 & 0.1218 & 0.4595 & 0.8631 & 0.1837 & 0.2471 \\ 0.5068 & 0.8627 & 0.8454 & 0.4403 & 0.7161 & 0.7578 & 0.9279 \\ 0.1450 & 0.1297 & 0.1770 & 0.2023 & 0.9684 & 0.6396 & 0.2200 \\ 0.6382 & 0.3370 & 0.4653 & 0.9578 & 0.6173 & 0.1641 & 0.9557 \end{bmatrix}$$

Test Problem A.7:

$$f(x) = e^{x_1 x_2 x_3 x_4 x_5} - 0.5(x_1^3 + x_2^3 + x_6^3 + 1)^2 + 2x_7 x_8$$

$$b^T = [0.2756, 0.9283, 0.7121, 0.0869, 0.9084, 0.4182]$$

$$\begin{bmatrix} 0.1561 & 0.1743 & 0.8339 & 0.0394 & 0.6963 & 0.0332 & 0.2050 & 0.2719 \\ 0.9810 & 0.8829 & 0.4494 & 0.9293 & 0.9583 & 0.9673 & 0.6076 & 0.5612 \\ 0.1322 & 0.8624 & 0.1449 & 0.3192 & 0.3495 & 0.7525 & 0.7416 & 0.8593 \\ 0.0870 & 0.9568 & 0.8332 & 0.0631 & 0.0539 & 0.0090 & 0.8811 & 0.0814 \\ 0.6490 & 0.6310 & 0.2537 & 0.9922 & 0.9580 & 0.9281 & 0.3140 & 0.6324 \\ 0.1774 & 0.5941 & 0.3514 & 0.2267 & 0.3360 & 0.8476 & 0.1344 & 0.2842 \end{bmatrix}$$

Test Problem A.8:

$$f(x) = (x_1 - 1)^2 + (x_7 - 1)^2 + 10\sum_{k=1}^{7}(10 - k)(x_k^2 - x_{k+1})^2$$

$$b^T = [0.1904, 0.3993, 0.7326, 0.1259, 0.5292, 0.4251, 0.8132]$$

$$\begin{bmatrix} 0.0435 & 0.0555 & 0.1488 & 0.2714 & 0.2047 & 0.6267 & 0.3194 & 0.2752 \\ 0.1495 & 0.4336 & 0.0635 & 0.6648 & 0.6100 & 0.2811 & 0.3213 & 0.9451 \\ 0.2675 & 0.1008 & 0.7718 & 0.0747 & 0.1574 & 0.7820 & 0.6325 & 0.8005 \\ 0.1245 & 0.1986 & 0.0332 & 0.0330 & 0.4692 & 0.0962 & 0.0637 & 0.0066 \\ 0.3358 & 0.2144 & 0.3903 & 0.4959 & 0.5513 & 0.1610 & 0.9640 & 0.0663 \\ 0.1220 & 0.3677 & 0.0197 & 0.8984 & 0.3886 & 0.7078 & 0.3703 & 0.3268 \\ 0.9448 & 0.8626 & 0.1840 & 0.8934 & 0.2138 & 0.6546 & 0.9129 & 0.9503 \end{bmatrix}$$

Appendix B

Test Problem B.1:

$f(x) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4$

$b^T = [0.2077, 0.4709, 0.8443]$

$$A = \begin{bmatrix} 0.4302 & 0.4464 & 0.0741 & 0.0751 \\ 0.1848 & 0.1603 & 0.4628 & 0.5929 \\ 0.9049 & 0.1707 & 0.8746 & 0.4210 \end{bmatrix}$$

Test Problem B.2:

$f(x) = x_1 - x_2 - x_3 - x_1x_3 + x_1x_4 + x_2x_3 - x_2x_4$

$b^T = [0.4228, 0.9427, 0.9831]$

$$\begin{bmatrix} 0.1280 & 0.7390 & 0.2852 & 0.2409 \\ 0.9991 & 0.7011 & 0.1688 & 0.9667 \\ 0.1711 & 0.6663 & 0.9882 & 0.6981 \end{bmatrix}$$

Test Problem B.3:

$f(x) = x_1x_2x_3x_4x_5$

$b^T = [0.6714, 0.5201, 0.1500]$

$$\begin{bmatrix} 0.4424 & 0.3592 & 0.6834 & 0.6329 & 0.9150 \\ 0.6878 & 0.7363 & 0.7040 & 0.6869 & 0.2002 \\ 0.6482 & 0.3947 & 0.4423 & 0.0769 & 0.0175 \end{bmatrix}$$

Test Problem B.4:

$f(x) = x_1 + 2x_2 + 4x_5 + e^{x_1x_4}$

$b^T = [0.6855, 0.5306, 0.5975, 0.2992]$

$$\begin{bmatrix} 0.1025 & 0.7780 & 0.3175 & 0.9357 & 0.7425 \\ 0.0163 & 0.2634 & 0.5542 & 0.4579 & 0.9213 \\ 0.7325 & 0.2481 & 0.8753 & 0.2405 & 0.4193 \\ 0.1260 & 0.2187 & 0.6164 & 0.7639 & 0.2962 \end{bmatrix}$$

Test Problem B.5:

$$f(x) = \sum_{k=1}^{6} [100(x_{k+1} - x_k^2)^2 + (1 - x_k)^2]$$

$b^T = [0.5846, 0.8277, 0.4425, 0.8266]$

$$\begin{bmatrix} 0.1187 & 0.4147 & 0.8051 & 0.3876 & 0.3643 & 0.7031 \\ 0.4761 & 0.8606 & 0.4514 & 0.0311 & 0.5323 & 0.1964 \\ 0.6618 & 0.2715 & 0.3826 & 0.0302 & 0.7117 & 0.1784 \\ 0.9081 & 0.1459 & 0.7896 & 0.9440 & 0.8715 & 0.1265 \end{bmatrix}$$

Test Problem B.6:

$f(x) = -0.5(x_1x_4 - x_2x_3 + x_2x_6 - x_5x_6 + x_5x_4 - x_6x_7)$

$b^T = [0.9879, 0.6321, 0.8082, 0.6650]$

$$\begin{bmatrix} 0.0832 & 0.3312 & 0.4580 & 0.7001 & 0.8287 & 0.9978 & 0.1876 \\ 0.3904 & 0.4277 & 0.2302 & 0.1373 & 0.4850 & 0.3495 & 0.8831 \\ 0.2393 & 0.8619 & 0.2734 & 0.8265 & 0.6598 & 0.4328 & 0.9315 \\ 0.4863 & 0.3787 & 0.6748 & 0.9301 & 0.4564 & 0.5893 & 0.8943 \end{bmatrix}$$

Test Problem B.7:

$f(x) = e^{x_1x_2x_3x_4x_5} - 0.5(x_1^3 + x_2^3 + x_6^3 + 1)^2$

$b^T = [0.9521, 0.0309, 0.8627, 0.8343, 0.6290]$

$$\begin{bmatrix} 0.9869 & 0.0805 & 0.8373 & 0.1417 & 0.9988 & 0.6320 \\ 0.0139 & 0.0169 & 0.0182 & 0.4379 & 0.0295 & 0.5095 \\ 0.2497 & 0.6914 & 0.8961 & 0.3504 & 0.8225 & 0.2433 \\ 0.9691 & 0.6170 & 0.5921 & 0.4785 & 0.5994 & 0.5714 \\ 0.6197 & 0.6298 & 0.2372 & 0.5874 & 0.2560 & 0.9817 \end{bmatrix}$$

Test Problem B.8:

$f(x) = (x_1 - 1)^2 + (x_7 - 1)^2 + 10 \sum_{k=1}^{7} (10 - k)(x_k^2 - x_{k+1})^2$

$b^T = [0.7840, 0.4648, 0.8864, 0.8352, 0.9839]$

$$\begin{bmatrix} 0.8522 & 0.2376 & 0.3586 & 0.7260 & 0.8891 & 0.2771 & 0.1316 \\ 0.4673 & 0.8176 & 0.1173 & 0.5350 & 0.1426 & 0.0020 & 0.2892 \\ 0.9707 & 0.4058 & 0.7248 & 0.1826 & 0.6193 & 0.8108 & 0.9630 \\ 0.8412 & 0.4663 & 0.7011 & 0.1124 & 0.6848 & 0.9434 & 0.4656 \\ 0.0785 & 0.9515 & 0.9997 & 0.0028 & 0.4982 & 0.6384 & 0.3852 \end{bmatrix}$$

References

[1] Chang, C. W., Shieh, B. S., Linear optimization problem constrained by fuzzy maxmin relation equations, Information Sciences 234 (2013) 7179.

[2] Chen, L. , Wang, P. P., Fuzzy relation equations (i): the general and specialized solving algorithms, Soft Computing 6 (5) (2002) 428-435.

[3] Chen, L., Wang, P. P., Fuzzy relation equations (ii): the branch-point-solutions and the categorized minimal solutions, Soft Computing 11 (2) (2007) 33-40.

[4] Dempe, S., Ruziyeva, A., On the calculation of a membership function for the solution of a fuzzy linear optimization problem, Fuzzy Sets and Systems 188 (2012) 58-67.

[5] Di Martino, F., Loia, V., Sessa, S., Digital watermarking in coding/decoding processes with fuzzy relation equations, Soft Computing 10(2006) 238-243.

[6] Di Nola, A., Sessa, S., Pedrycz, W., Sanchez, E., Fuzzy relational equations and their applications in knowledge engineering, Dordrecht: Kluwer academic press, 1989.

[7] Dubey, D., Chandra, S., Mehra, A., Fuzzy linear programming under interval uncertainty based on IFS representation, Fuzzy Sets and Systems 188 (2012) 68-87.

[8] Dubois, D., Prade, H., Fundamentals of Fuzzy Sets, Kluwer, Boston, 2000.

[9] Fan, Y. R., Huang, G. H., Yang, A. L., Generalized fuzzy linear programming for decision making under uncertainty: Feasibility of fuzzy solutions and solving approach, Information Sciences 241 (2013) 12-27.

[10] Fang, S. C., Li, G., Solving fuzzy relation equations with a linear objective function, Fuzzy Sets and Systems 103(1999) 107-113.

[11] Fernandez, M. J., Gil, P., Some specific types of fuzzy relation equations, Information Sciences 164 (2004) 189-195.

[12] Freson, S., De Baets, B., De Meyer, H., Linear optimization with bipolar maxmin constraints, Information Sciences 234 (2013) 315.

[13] Ghodousian, A., Khorram, E., An algorithm for optimizing the linear function with fuzzy relation equation constraints regarding max-prod composition, Applied Mathematics and Computation 178 (2006) 502-509.

[14] Ghodousian, A., Optimization of the reducible objective functions with monotone factors subject to FRI constraints defined with continuous t-norms, Archives of Industrial Engineering 1(1) (2018) 1-19.

[15] Ghodousian, A., Babalhavaeji, A., An efficient genetic algorithm for solving nonlinear optimization problems defined with fuzzy relational equations and max-Lukasiewicz composition, Applied Soft Computing 69 (2018) 475492.

[16] Ghodousian, A., Naeeimib, M., Babalhavaeji, A., Nonlinear optimization problem subjected to fuzzy relational equations defined by Dubois-Prade family of t-norms, Computers and Industrial Engineering 119 (2018) 167180.

[17] Ghodousian, A., Raeisian Parvari, M., A modified PSO algorithm for linear optimization problem subject to the generalized fuzzy relational inequalities with fuzzy constraints (FRI-FC), Information Sciences 418419 (2017) 317345.

[18] Ghodousian, A., Zarghani, R., Linear optimization on the intersection of two fuzzy relational inequalities defined with Yager family of t-norms, Journal of Algorithms and Computation 49 (1) (2017) 55 82.

[19] Ghodousian, A., Ahmadi, A., Dehghani, A., Solving a non-convex non-linear optimization problem constrained by fuzzy relational equations and Sugeno-Weber family of t-norms, Journal of Algorithms and Computation 49 (2) (2017) 63 101.

[20] Ghodousian, A., Khorram, E., Fuzzy linear optimization in the presence of the fuzzy relation inequality constraints with max-min composition, Information Sciences 178 (2008) 501-519.

[21] Ghodousian, A., Khorram, E., Linear optimization with an arbitrary fuzzy relational inequality, Fuzzy Sets and Systems 206 (2012) 89-102.

[22] Ghodousian, A., Khorram, E., Solving a linear programming problem with the convex combination of the max-min and the max-average fuzzy relation equations, Applied Mathematics and computation 180 (2006) 411-418.

[23] Guo, F.F., Pang, L. P., Meng, D., Xia, Z. Q., An algorithm for solving optimization problems with fuzzy relational inequality constraints, Information Sciences 252 (2013) 20-31.

[24] Guo, F. F., Xia, Z. Q., An algorithm for solving optimization problems with one linear objective function and finitely many constraints of fuzzy relation inequalities, Fuzzy Optimization and Decision Making 5(2006) 33-47.

[25] Guu, S. M., Wu, Y. K., Minimizing a linear objective function under a max-t-norm fuzzy relational equation constraint, Fuzzy Sets and Systems 161 (2010) 285-297.

[26] Guu, S. M., Wu, Y. K., Minimizing a linear objective function with fuzzy relation equation constraints, Fuzzy Optimization and Decision Making 12 (2002) 1568-4539.

[27] Guu, S. M., Wu, Y. K., Minimizing an linear objective function under a max-t-norm fuzzy relational equation constraint, Fuzzy Sets and Systems 161 (2010) 285-297.

[28] Han, S. C., Li, H. X., Notes on pseudo-t-norms and implication operators on a complete Brouwerian lattice and pseudo-t-norms and implication operators: direct products and direct product decompositions, Fuzzy Sets and Systems 153(2005) 289-294.

[29] Hock, W., Schittkowski, K., Test examples for nonlinear programming codes Lecture Notes in Economics and Mathematical Systems, vol. 187, Springer, New York, 1981.

[30] Khorram, E., Ghodousian, A., Linear objective function optimization with fuzzy relation equation constraints regarding max-av composition, Applied Mathematics and Computation 173 (2006) 872-886.

[31] Klement, E. P., Mesiar, R., Pap, E., Triangular norms. Position paper I: Basic analytical and algebraic properties, Fuzzy Sets and Systems 143(2004) 5-26.

[32] Lee, H. C., Guu, S. M., On the optimal three-tier multimedia streaming services, Fuzzy Optimization and Decision Making 2(1) (2002) 31-39.

[33] Li, P. K., Fang, S. C., On the resolution and optimization of a system of fuzzy relational equations with sup-t composition, Fuzzy Optimization and Decision Making 7 (2008) 169-214.

[34] Li, P., Liu, Y., Linear optimization with bipolar fuzzy relational equation constraints using lukasiewicz triangular norm, Soft Computing 18 (2014) 1399-1404.

[35] Li, P., Fang, S. C., A survey on fuzzy relational equations, part i: classification and solvability, Fuzzy Optimization and Decision Making 8 (2009) 179-229.

[36] Lin, J. L., On the relation between fuzzy max-archimedean t-norm relational equations and the covering problem, Fuzzy Sets and Systems 160 (2009) 2328-2344.

[37] Lin, J. L., Wu, Y. K., Guu, S. M., On fuzzy relational equations and the covering problem, Information Sciences 181 (2011) 2951-2963.

[38] Liu, C.C., Lur, Y.Y., Wu, Y. K., Linear optimization of bipolar fuzzy relational equations with max-ukasiewicz composition, Information Sciences 360 (2016) 149162.

[39] Loetamonphong, J., Fang, S. C., Optimization of fuzzy relation equations with max-product composition, Fuzzy Sets and Systems 118 (2001) 509-517.

[40] Loia, V., Sessa, S., Fuzzy relation equations for coding/decoding processes of images and videos, Information Sciences 171(2005) 145-172.

[41] Lu, J., Fang, S. C., Solving nonlinear optimization problems with fuzzy relation equations constraints, Fuzzy Sets and Systems 119(2001) 1-20.

[42] Markovskii, A. V., On the relation between equations with max-product composition and the covering problem, Fuzzy Sets and Systems 153 (2005) 261-273.

[43] Nayeed, F., Hanmandlu, M., Three information set-based feature types for the recognition of faces / Signal, Image and Video Processing 10(2) (2016) 327-334.

[44] Nobuhara, H., Hirota, K., Pedrycz, W., Relational image compression: optimizations through the design of fuzzy coders and YUV colors space, Soft Computing 9 (2005) 471-479.

[45] Nobuhara, H., Hirota, K., Di Martino, F., Pedrycz, W., Sessa, S., Fuzzy relation equations for compression/decompression processes of color images in the RGB and YUV color spaces, Fuzzy Optimization and Decision Making 4 (2005) 235-246.

[46] Oantarci, S., Vahaplar, A., Kinay, A. O., Nasibov, E., Influence of T-norm and T-conorm operators in Fuzzy ID3 algorithm , IEEE International Conference on Fuzzy Systems, 2015, Istanbul, Turkey.

[47] Oastro-Company, F., Tirado, P., On Yager and Hamacher t-Norms and Fuzzy Metric Spaces , International Journal of Intelligent Systems 29 (12) (2014) 1173-1180.

[48] Pedrycz, W., Vasilakos, A. V., Modularization of fuzzy relational equations, Soft Computing 6(2002) 3-37.

[49] Peeva, K., Resolution of fuzzy relational equations-methods, algorithm and software with applications, Information Sciences 234 (2013) 44-63.

[50] Perfilieva, I., Fuzzy function as an approximate solution to a system of fuzzy relation equations, Fuzzy Sets and Systems 147(2004) 363-383.

[51] Perfilieva, I., Novak, V., System of fuzzy relation equations model of IF-THEN rules, Information Sciences 177 (16) (2007) 3218-3227.

[52] Perfilieva, I., Finitary solvability conditions for systems of fuzzy relation equations, Information Sciences 234 (2013)29-43.

[53] Qu, X. B., Wang, X. P., Man-hua. H. Lei, , Conditions under which the solution sets of fuzzy relational equations over complete Brouwerian lattices form lattices, Fuzzy Sets and Systems 234 (2014) 34-45.

[54] Sanchez, E., Resolution of composite fuzzy relation equations, Inf. Control 30(1976) 38-48.

[55] Shieh, B. S., Infinite fuzzy relation equations with continuous t-norms, Information Sciences 178 (2008) 1961-1967.

[56] Shieh, B. S., Minimizing a linear objective function under a fuzzy max-t-norm relation equation constraint, Information Sciences 181 (2011) 832-841.

[57] Sun, F., Conditions for the existence of the least solution and minimal solutions to fuzzy relation equations over complete Brouwerian lattices, Information Sciences 205 (2012) 86-92.

[58] Sun, F., Wang, X. P., Qu, x. B., Minimal join decompositions and their applications to fuzzy relation equations over complete Brouwerian lattices, Information Sciences 224 (2013) 143-151.

[59] Wang, S., Fang, S. C., Nuttle, H. L. W., Solution sets of interval-valued fuzzy relational equations, Fuzzy Optimization and Decision Making 2 (1)(2003) 41-60.

[60] Wang, P. Z., Latticized linear programming and fuzzy relation inequalities, Journal of Mathematical Analysis and Applications 159(1991) 72-87.

[61] Wu, Y. K., Guu, S. M., A note on fuzzy relation programming problems with max-strict-t-norm composition, Fuzzy Optimization and Decision Making 3(2004) 271-278.

[62] Wu, Y. K., Guu, S. M., An efficient procedure for solving a fuzzy relation equation with max-Archimedean t-norm composition, IEEE Transactions on Fuzzy Systems 16 (2008) 73-84.

[63] Wu, Y. K., Optimization of fuzzy relational equations with max-av composition, Information Sciences 177 (2007) 4216-4229.

[64] Wu, Y. K., Guu, S. M., Minimizing a linear function under a fuzzy max-min relational equation constraints, Fuzzy Sets and Systems 150 (2005) 147-162.

[65] Wu, Y. K., Guu, S. M., Liu, J. Y., Reducing the search space of a linear fractional programming problem under fuzzy relational equations with max-Archimedean t-norm composition, Fuzzy Sets and Systems 159 (2008) 3347-3359.

[66] Xiong, Q.Q., Wang, X. P., Fuzzy relational equations on complete Brouwerian lattices, Information Sciences 193 (2012) 141-152.

[67] Xiong, Q. Q., Wang, X. P., Some properties of sup-min fuzzy relational equations on infinite domains, Fuzzy Sets and Systems 151(2005) 393-402.

[68] Yang, X. P., Zhou, X. G., Cao, B. Y., Latticized linear programming subject to max-product fuzzy relation inequalities with application in wireless communication, Information Sciences 358359 (2016) 4455.

[69] Yang, S. J., An algorithm for minimizing a linear objective function subject to the fuzzy relation inequalities with addition-min composition, Fuzzy Sets and Systems 255 (2014) 41-51.

[70] Yeh, C. T., On the minimal solutions of max-min fuzzy relation equations, Fuzzy Sets and Systems 159 (2008) 23-39.