AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 9(6): 13624–13641.
DOI: 10.3934/math.2024665
Received: 31 December 2023
Revised: 06 February 2024
Accepted: 18 March 2024
Published: 12 April 2024

Research article

Weighted minimax programming subject to the max-min fuzzy relation inequalities

Miaoxia Chen[1,2], Abdul Samad Shibghatullah[1], Kasthuri Subramaniam[1] and Xiaopeng Yang[1,3,*]

[1] Institute of Computer Science & Digital Innovation, UCSI University, Kuala Lumpur, 56000, Malaysia

[2] Department of Education, Hanshan Normal University, Chaozhou, 521041, China

[3] School of Mathematics and Statistics, Hanshan Normal University, Chaozhou, 521041, China

* **Correspondence:** Email: 706697032@qq.com; Tel: +8615976337822.

Abstract: Recently, max-min fuzzy relation inequalities (FRIs) have been used to model a (peer-to-peer) P2P network system. Any feasible scheme in the P2P network system is reflected by a solution of the max-min FRIs. One of the objectives of system managers is to decrease network congestion. To satisfy this objective, we attempt to minimize a weighted minimax function motivated by existing research. As a consequence, we establish a weighted minimax programming model in which the constraint is the max-min FRIs. Our goal in this work is to develop an effective algorithm to obtain the optimal solution of the optimization model. The so-called SCP-based algorithm is proposed to find the optimal solution. A numerical example shows the efficiency of our proposed SCP-based algorithm.

Keywords: fuzzy relation inequality; max-min composition; P2P network system; weighted minimax programming; mathematical model
Mathematics Subject Classification: 90C70, 90C90

1. Introduction

In recent years, the addition-min fuzzy relation inequality (FRI) was first introduced in [1, 2] for describing the flow constraints in the peer-to-peer (P2P) network system. Since the minimal solutions play a key role in constructing the complete solution set for the addition-min FRIs [3–5], Li et al. [1] proposed a feasible approach to find some specific minimal solutions. However, as shown in [2], the addition-min FRIs usually have an infinite number of minimal solutions, and it is difficult to determine all the minimal solutions. To obtain specific minimal solutions, various approaches have been developed. In [6], the lexicographic minimum solution was defined and studied. An effective resolution algorithm and some illustrative numerical examples were provided. The concept of the lexicographic minimum solution was also extended to random-term-absent addition-min FRIs [7]. It can be formally proven that any lexicography minimum solution should also be a minimal solution in the addition-min FRI system. In [8], Li et al. investigated another kind of minimal solution for addition-min FRIs. The authors attempted to find a minimal solution that was less than or equal to a given solution [8]. Moreover, to obtain specific minimal solutions, solving an optimization problem is also an effective approach. For instance, the optimization problem with a linear objective function was studied in [9, 10], with addition-min FRIs constraints. Considering the fairness among the terminals in the P2P network system, Yang et al. [11] and Chiu et al. [12] further investigated fuzzy relation minimax programming with addition-min composition. However, it was shown that its optimal solutions were usually nonunique. Thus, Wu et al. [13] and Yang et al. [14] further searched for the minimal optimal solutions. By adding some weighted factors to the terminals, the corresponding fuzzy relation weighted minimax programming was also investigated [15–17].

In references [1–17], addition-min FRIs were introduced to investigate the P2P network system. As noted in [18–21], by applying the addition-min FRIs to model the P2P network system, the authors considered the total download speed of each terminal and downloaded its requested data from other terminals. However, it was also indicated in [18–21] that, in some cases, the highest download speed should be considered. Next, we describe the requirements of the highest download speed for the terminals in a P2P network system. Similarly, it is assumed that there are n terminals in the system, represented by $T_1, T_2, \cdots, T_n$ (see Figure 1). After accepting the downloading request from other terminals, we suppose that terminal T_j transmits its local file to any other terminal at quality level t_j. If the bandwidth between T_i and T_j (exactly from T_j to T_i) is a_{ij} (see Figure 2), then the actual download traffic of T_i from T_j is $a_{ij} \wedge t_j$.

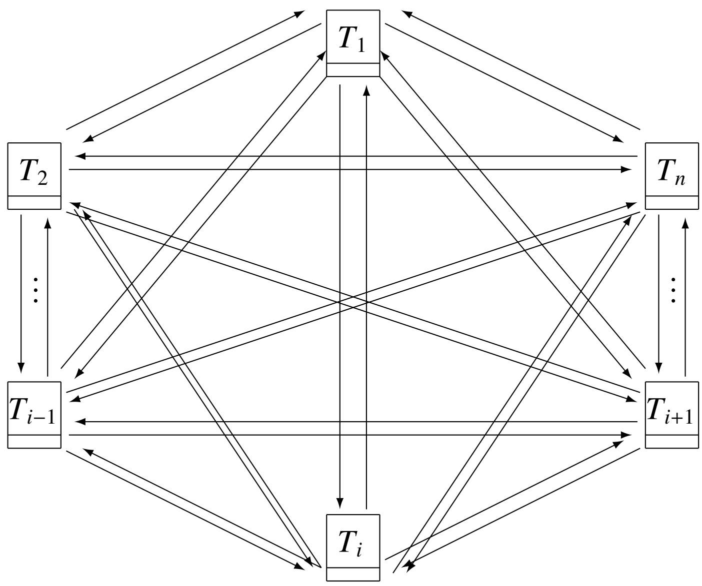

Figure 1. The P2P (Peer-to-Peer) network system.

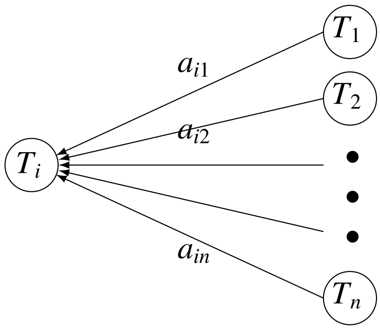

Figure 2. The bandwidths between T_i and the other terminals.

Considering the total download traffic of each terminal, the P2P network system can be modeled by the addition-min FRIs. However, in some cases, the specific file should be downloaded from a single terminal. In such cases, T_i downloads its requested file from the terminal with the highest download traffic, i.e., $a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n$. Let us suppose that the requirement of the highest download traffic of T_i is no less than b_i, $i = 1, 2, \cdots, m$. Then the requirements of all the terminals in a P2P network system should be modeled by the following max-min FRIs:

$$a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \geq b_i, \quad i = 1, 2, \cdots, m. \tag{1.1}$$

Moreover, if the requirement of each terminal for the highest download speed has both an upper bound and a lower bound, then the corresponding max-min FRIs should be further written as

$$b_i \leq a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \leq d_i, \quad i = 1, 2, \cdots, m. \tag{1.2}$$

For inconsistent system (1.1), Yang established an evaluation model for a given vector, based on which the approximate solution was defined and investigated [22]. For the consistent system (1.1), Zhong et al. [23] focused on the lexicographic minimum solution. Xiao et al. investigated the evaluation and derived a classification of system solutions (1.1) [24]. A resolution algorithm was developed with an illustrative example. Moreover, the lexicographic minimum solution was also introduced for the above system (1.2) [25]. Then considering the stability of the P2P network system, Chen et al. [26] defined the concept of interval solutions for system (1.2). The authors proposed an effective algorithm to find the so-called widest interval solution of system (1.2) in [26].

As mentioned above, with the constraint system of addition-min FRIs, weighted minimax programming, or even minimax programming as its specific form, has been studied, providing some efficient resolution algorithms [11–17]. However, when considering the highest download speed, the relevant weighted minimax programming has not been studied for the corresponding max-min system (1.2). Consequently, we focus on such an optimization problem in this work. The formulistic form of the weighted minimax programming subject to system (1.2) is

$$\begin{aligned} \min \quad & z(t) = c_1 t_1 \vee c_2 t_2 \vee \cdots \vee c_n t_n, \\ \text{s.t.} \quad & \begin{cases} b_i \leq a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \leq d_i, \\ i = 1, 2, \cdots, m. \end{cases} \end{aligned} \tag{1.3}$$

As shown in [11–17], in system (1.2) or problem (1.3), t_j (measure: Mbps) represents the quality level on which the jth terminal shares (sends out) its local resources with the other terminals. To decrease network congestion, system managers usually try to minimize the values of $t_1, t_2, \cdots, t_n$. However, in most cases, these values cannot be minimized simultaneously. Instead of minimizing all these variables, minimizing a specific function with the variables $t_1, t_2, \cdots, t_n$ is much more realistic. For instance, in problem (1.3), we adopt the weighted minimax function. In such a weighted minimax function in (1.3), the parameter c_j represents the weighted factor of the jth terminal in the P2P network system. In this work, we design several effective algorithms for identifying the optimal solution.

In summary, the innovations and contributions of this work can be summarized as follows:
(i) Instead of the total download traffic of the terminal, we consider the highest download traffic. Consequently, the corresponding max-min FRIs are employed.
(ii) To decrease network congestion in the P2P network system, we consider a given weighted factor for each terminal. Moreover, we further establish a relevant weighted minimax programming problem with max-min FRI constraints.
(iii) To solve our established weighted minimax optimization model, we propose the so-called single-constraint programming approach for identifying the optimal solution.

The following content is organized as follows. In Section 2, we describe some foundational concepts and results on max-min FRIs, i.e., system (1.2). Our major results are presented in Section 3. In this section, we introduce the single-constraint programming (SCP) approach for addressing our studied problem (1.3). The original problem (1.3) is separated into several subproblems and solved. Moreover, our proposed approach is a step-by-step SCP-based algorithm. In Section 4, a detailed numerical example is provided to demonstrate the SCP-based algorithm. In Section 5, we compare our problem and the proposed method to the existing ones. Section 6 provides a simple conclusion.

2. On the max-min fuzzy relation inequalities

In this section, we provide some foundational results on the max-min fuzzy relation inequalities, i.e., system (1.2). These existing results are helpful for the resolution of problem (1.3).

Let us denote

$$A = (a_{ij})_{m\times n}, t = (t_1, \cdots, t_n), b = (b_1, \cdots, b_m), d = (d_1, \cdots, d_m).$$

Then we represent system (1.2) as

$$b \le A \circ t \le d. \tag{2.1}$$

Moreover, the solution set of system (1.2) is indeed

$$\mathcal{T}^{A,b,d} = \{t \in [0, 1]^n | b \le A \circ t \le d\}.$$

For convenience, we let

$$\mathcal{I} = \{1, 2, \cdots, m\}, \ \ \mathcal{J} = \{1, 2, \cdots, n\}.$$

Definition 1. (Consistent; Maximum solution [25]) System (1.2) is said to be consistent when its solution set is nonempty, i.e., $\mathcal{T}^{A,b,d} \neq \emptyset$; otherwise, system (1.2) is inconsistent. When system (1.2) is consistent and there exists a solution $\bar{t} \in \mathcal{T}^{A,b,d}$ such that $\bar{t} \ge t$ for any $t \in \mathcal{T}^{A,b,d}$, then we say $\bar{t}$ is the maximum solution of (1.2).

In system (1.2), we construct the vector $\hat{t} = (\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_n)$ as follows:

$$\hat{t}_j = \begin{cases} \bigwedge_{i \in \mathcal{I}_j} d_i, & \text{if } \mathcal{I}_j \neq \emptyset, \\ 1, & \text{if } \mathcal{I}_j = \emptyset, \end{cases} \tag{2.2}$$

where $\mathcal{I}_j = \{i \in \mathcal{I} | a_{ij} > d_i\}$, $\forall j \in \mathcal{J}$. The vector $\hat{t}$ is important to determine whether system (1.2) is consistent.

Proposition 1. *[25] Let $t \in \mathcal{T}^{A,b,d}$ be a solution of system* (1.2)*, and the vector $\hat{t}$ is defined by* (2.2)*. Then, we have $t \leq \hat{t}$.*

Theorem 1. *[25] System* (1.2) *is consistent, i.e., $\mathcal{T}^{A,b,d} \neq \emptyset$, if and only if $\hat{t} \in \mathcal{T}^{A,b,d}$.*

According to Proposition 1 and Theorem 1, we know that when $\mathcal{T}^{A,b,d} \neq \emptyset$, $\hat{t}$ should be the unique maximum solution of system (1.2). The potential maximum solution $\hat{t}$ can be used to check the consistency of (1.2).

Definition 2. (Minimal solution [25]) Let (1.2) be consistent and $\check{t} \in \mathcal{T}^{A,b,d}$. We say that $\check{t}$ is a minimal solution if there is no $t \in \mathcal{T}^{A,b,d}$ such that $t \leq \check{t}$ and $t \neq \check{t}$.

The conservative path approach was proposed in [27, 28] to obtain all the minimal solutions of the system (1.2). Moreover, it was shown that system (1.2) has a finite number of minimal solutions when it is consistent. For system (1.2), we denote the set of all minimal solutions by

$$\check{\mathcal{T}}^{A,b,d} = \{\check{t} \in \mathcal{T}^{A,b,d} | \check{t} \text{ is a minimal solution}\}.$$

Based on the maximum solution $\hat{t}$ and the minimal solution set $\check{\mathcal{T}}^{A,b,d}$, the complete solution set to (1.2) can be characterized by Theorem 2.

Theorem 2. *[25] When system* (1.2) *is consistent, the solution set $\mathcal{T}^{A,b,d}$ is*

$$\mathcal{T}^{A,b,d} = \bigcup_{\check{t} \in \check{\mathcal{T}}^{A,b,d}} [\check{t}, \hat{t}]. \tag{2.3}$$

Since all minimal solutions can be found by the conservative path approach [27, 28], one can obtain the complete solution set $\mathcal{T}^{A,b,d}$ for system (1.2).

3. Resolution of problem (1.3)

3.1. Existence of the optimal solution and the SCP-based approach to solve problem (1.3)

In this subsection, we first illustrate the existence of the optimal solution for problem (1.3). The optimal solution exists if and only if the feasible domain, i.e., $\mathcal{T}^{A,b,d}$, is nonempty. Afterward, we assume $\mathcal{T}^{A,b,d} \neq \emptyset$, and we attempt to solve problem (1.3). The original problem (1.3) is separated into m subproblems according to the constraints, i.e., system (1.2). Each subproblem has the same objective function as problem (1.3) and a single inequality in the constraint. Thus, each subproblem is indeed a single-constraint programming (SCP) problem. The optimal solution of problem (1.3) can be generated by the optimal solutions of the subproblems. Since problem (1.3) is solved via single-constraint programming, we refer to our resolution method as the SCP-based approach.

Theorem 3. *(Existence of the optimal solution) System* (1.2) *is consistent, i.e.,* $\mathcal{T}^{A,b,d} \neq \emptyset$*, if and only if problem* (1.3) *has at least one optimal solution.*

Proof. It has been shown previously that system (1.2) has a finite number of minimal solutions since it is consistent. Without loss of generality, we denote the minimal solution set by

$$\check{\mathcal{T}}^{A,b,d} = \{\check{t}^1, \check{t}^2, \cdots, \check{t}^s\}.$$

The objective function values of $\{\check{t}^1, \check{t}^2, \cdots, \check{t}^s\}$ can be found as $\{z(\check{t}^1), z(\check{t}^2), \cdots, z(\check{t}^s)\}$. Let us denote

$$z^* = \min\{z(\check{t}^1), z(\check{t}^2), \cdots, z(\check{t}^s)\}. \tag{3.1}$$

Moreover, there exists $l^* \in \{1, 2, \cdots, s\}$ such that $z(\check{t}^{l^*}) = z^*$, and

$$z^* \leq z(\check{t}^l), \quad \forall l \in \{1, 2, \cdots, s\}. \tag{3.2}$$

Obviously, the minimal solution $\check{t}^{l^*}$ is also a feasible solution to problem (1.3). Next, we further verified that it is an optimal solution.

Let $t \in \mathcal{T}^{A,b,d}$ be an arbitrary solution of (1.2). According to Theorem 2, $l' \in \{1, 2, \cdots, s\}$ and $\check{t}^{l'} \in \check{\mathcal{T}}^{A,b,d}$, such that $t \in [\check{t}^{l'}, \hat{t}]$, i.e., $\check{t}^{l'} \leq t \leq \hat{t}$. Thus,

$$z(\check{t}^{l'}) = c_1\check{t}_1^{l'} \vee \cdots \vee c_n\check{t}_n^{l'} \leq c_1t_1 \vee \cdots \vee c_nt_n = z(t). \tag{3.3}$$

Let us note that $l' \in \{1, 2, \cdots, s\}$. The inequalities (3.2) and (3.3) imply that $z^* \leq z(t)$, i.e., $z(\check{t}^{l^*}) \leq z(t)$. As a result, $\check{t}^{l^*}$ is an optimal solution of (1.3). □

It has been shown in Theorem 3 that the optimal solution of problem (1.3) should exist when system (1.2) is consistent. Next, we consider the resolution of problem (1.3) with the assumption that $\mathcal{T}^{A,b,d} \neq \emptyset$.

Based on the maximum solution $\hat{t}$ and the formulae in problem (1.3), we construct the following single-constraint programming problem:

$$\begin{aligned} (P_i)\ \min\ \ & z(t) = c_1t_1 \vee c_2t_2 \vee \cdots \vee c_nt_n, \\ \text{s.t.}\ \ & \begin{cases} b_i \leq a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \leq d_i, \\ t \leq \hat{t}, \end{cases} \end{aligned} \tag{3.4}$$

for each $i \in \mathcal{I}$. Then, we obtain m subproblems, denoted by $\{(P_1), (P_2), \cdots, (P_m)\}$, corresponding to the original problem (1.3).

In fact, the optimal solution of problem (1.3) can be generated by the optimal solutions of these subproblems, as indicated in what follows. Next, we provide an algorithm to obtain the optimal solution of each subproblem, and then we show the relationship between the original problem (1.3) and the subproblems $\{(P_1), (P_2), \cdots, (P_m)\}$.

3.2. Algorithm for obtaining an optimal solution to subproblem (P_i)

Let $i \in \mathcal{I}$ be an arbitrary index in $\mathcal{I}$. In this subsection, we propose an effective approach to obtain an optimal solution of subproblem (P_i), i.e., problem (3.4). In the remainder of this subsection, we always assume that i is a given index.

We denote the index set as

$$\mathcal{J}_i^{\hat{t}} = \{j \in \mathcal{J} | a_{ij} \geq b_i, \hat{t}_j \geq b_i\}. \tag{3.5}$$

It is clear that $j \in \mathcal{J}_i^{\hat{t}}$ if and only if $a_{ij} \wedge \hat{t}_j \geq b_i$.

Proposition 2. *$\hat{t}$ is a solution of system* (1.2)*, if and only if $\mathcal{J}_k^{\hat{t}} \neq \emptyset$ holds for any $k \in \mathcal{I}$.*

Proof. (⇒) If $\hat{t}$ is a solution of system (1.2), we have

$$a_{k1} \wedge \hat{t}_1 \vee a_{k2} \wedge \hat{t}_2 \vee \cdots \vee a_{kn} \wedge \hat{t}_n \geq b_k, \forall k \in \mathcal{I}. \tag{3.6}$$

Hence, there exists $j' \in \mathcal{J}$ such that $a_{kj'} \wedge \hat{t}_{j'} \geq b_k$. This implies that $j' \in \mathcal{J}_k^{\hat{t}}$, i.e., $\mathcal{J}_k^{\hat{t}} \neq \emptyset$.

(⇐) If $\mathcal{J}_k^{\hat{t}} \neq \emptyset$, $\forall k \in \mathcal{I}$, then there exists $j_k \in \mathcal{J}_k^{\hat{t}}$ for each k. Since $j_k \in \mathcal{J}_k^{\hat{t}} \subseteq \mathcal{J}$, by (3.5) we have

$$a_{k1} \wedge \hat{t}_1 \vee a_{k2} \wedge \hat{t}_2 \vee \cdots \vee a_{kn} \wedge \hat{t}_n \geq a_{kj_k} \wedge \hat{t}_{j_k} \geq b_k, \forall k \in \mathcal{I}. \tag{3.7}$$

Therefore, $\hat{t}$ is a solution of system (1.2). □

According to Theorem 1 and Proposition 2, we find Corollary 1.

Corollary 1. *$\mathcal{T}^{A,b,d} \neq \emptyset$ if and only if $\mathcal{J}_k^{\hat{t}} neq \emptyset$ holds for any $k \in \mathcal{I}$.*

Obviously, Corollary 1 can also be used to check the consistency of system (1.2) through the index sets $\{\mathcal{J}_1^{\hat{t}}, \mathcal{J}_2^{\hat{t}}, \cdots, \mathcal{J}_m^{\hat{t}}\}$.

Based on the index set $\mathcal{J}_i^{\hat{t}}$, defined by (3.5), we find the optimal index as

$$p_i^* = \arg \min_{j \in \mathcal{J}_i^{\hat{t}}} \{c_j\}. \tag{3.8}$$

Then we have $p_i^* \in \mathcal{J}_i^{\hat{t}}$ and $c_{p_i^*} = \min\{c_j | j \in \mathcal{J}_i^{\hat{t}}\}$. Furthermore, we construct the vector $t^{*i} = (t_1^{*i}, t_2^{*i}, \cdots, t_n^{*i})$ as

$$t_j^{*i} = \begin{cases} b_i, & \text{if } j = p_i^*, \\ 0, & \text{if } j \neq p_i^*. \end{cases} \tag{3.9}$$

We show that the vector t^{*i} is indeed an optimal solution of the subproblem (P_i).

Theorem 4. *(Optimal solution of (P_i)) Let us suppose $\mathcal{T}^{A,b,d} \neq \emptyset$. Then, the vector t^{*i} defined above is an optimal solution of the subproblem (P_i).*

Proof. (Feasibility) Let $j' = p_i^* \in \mathcal{J}_i^{\hat{t}}$. By (3.5), we have

$$a_{ij'} \geq b_i, \ \ \hat{t}_{j'} \geq b_i. \tag{3.10}$$

By (3.9), we also have

$$t_{j'}^{*i} = b_i, \tag{3.11}$$

and

$$t_j^{*i} = 0, \ \ \forall j \neq j', j \in \mathcal{J}. \tag{3.12}$$

Since $a_{ij'} \geq b_i$, it is immediate that $a_{ij'} \wedge b_i = b_i$. Considering (3.12), we have

$$a_{i1} \wedge t_1^{*i} \vee \cdots \vee a_{in} \wedge t_n^{*i} = a_{ij'} \wedge b_i = b_i. \tag{3.13}$$

According to (3.10)–(3.12), we have
$$\hat{t}_{j'} \geq b_i = t^{*i}_{j'}, \tag{3.14}$$
and
$$\hat{t}_j \geq 0 = t^{*i}_j, \ \ \forall j \neq j', j \in \mathcal{J}. \tag{3.15}$$
Formulae (3.14) and (3.15) imply that $t^{*i} \leq \hat{t}$. Combining formula (3.13), t^{*i} is a feasible solution of subproblem (P_i).

According to (3.11) and (3.12), $z(t^{*i}) = c_{j'}b_i$. Let us take an arbitrary feasible solution of (1.3) as t. Then, it holds that
$$\begin{cases} b_i \leq a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \leq d_i, \\ t \leq \hat{t}. \end{cases} \tag{3.16}$$
There exists $j'' \in \mathcal{J}$ such that $a_{ij''} \wedge t_{j''} \geq b_i$, i.e., $a_{ij''} \geq b_i$ and $t_{j''} \geq b_i$. By (3.5), we have $j'' \in \mathcal{J}^{\hat{t}}_i$. Since $j' = p^*_i$, by (3.8) we have
$$c_{j'} = c_{p^*_i} = \min\{c_j | j \in \mathcal{J}^{\hat{t}}_i\} \leq c_{j''}. \tag{3.17}$$
However, considering $t_{j''} \geq b_i$, we have
$$z(t) = c_1 t_1 \vee \cdots \vee c_n t_n \geq c_{j''} t_{j''} \geq c_{j'} b_i = z(t^{*i}). \tag{3.18}$$

As a consequence, t^{*i} is an optimal solution of subproblem (P_i). □

Summarizing the above results, we develop Algorithm I to solve subproblem (P_i).

Algorithm I: To calculate an optimal solution of the subproblem (P_i)

Step 1: Construct the vector $\hat{t} = (\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_n)$ following (2.2).

Step 2: Apply the vector $\hat{t}$; check the consistency of system (1.2) following Theorem 1. If $\hat{t} \in \mathcal{T}^{A,b,d}$, then $\mathcal{T}^{A,b,d} \neq \emptyset$ and problem (1.3) is solvable. Continue to the next step. Otherwise, problem (1.3) is unsolvable; stop.

Step 3: Find the index set $\mathcal{J}^{\hat{t}}_i$ according to (3.5).

Step 4: Find the optimal index p^*_i according to $\mathcal{J}^{\hat{t}}_i$ and (3.8).

Step 5: Find the vector $t^{*i} = (t^{*i}_1, t^{*i}_2, \cdots, t^{*i}_n)$ according to the optimal indices p^*_i and (3.9). Then, by Theorem 4, t^{*i} is an optimal solution of subproblem (P_i).

Example 1. Let us consider the following single-constraint programming problem:
$$\begin{aligned} (P_0) \ \min \ \ & z(t) = 0.4t_1 \vee 0.5t_2 \vee 0.7t_3 \vee 0.3t_4 \vee 0.5t_5 \vee 0.6t_6, \\ \text{s.t.} \ \ & \begin{cases} 0.38 \leq 0.9 \wedge t_1 \vee 0.6 \wedge t_2 \vee 0.8 \wedge t_3 \vee 0.2 \wedge t_4 \vee 0.3 \wedge t_5 \vee 0.4 \wedge t_6 \leq 0.76, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases} \end{aligned} \tag{3.19}$$
Algorithm I is applied to find an optimal solution to problem (3.19), i.e., problem (P_0).

Solution. **Steps 1 and 2:** The constraint is extracted from system (4.1) appearing in Example 2 below. We verify that system (4.1) is consistent, with the maximum solution $\hat{t} = (0.76, 0.77, 0.76, 0.75, 0.85, 1)$. Hence, we go directly to Step 3.

Step 3: Since

$$\begin{cases} 0.9 \wedge \hat{t}_1 = 0.9 \wedge 0.76 = 0.76 > 0.38, \\ 0.6 \wedge \hat{t}_2 = 0.6 \wedge 0.77 = 0.6 > 0.38, \\ 0.8 \wedge \hat{t}_3 = 0.8 \wedge 0.76 = 0.76 > 0.38, \\ 0.2 \wedge \hat{t}_4 = 0.2 \wedge 0.75 = 0.2 < 0.38, \\ 0.3 \wedge \hat{t}_5 = 0.3 \wedge 0.85 = 0.3 < 0.38, \\ 0.4 \wedge \hat{t}_6 = 0.4 \wedge 1 = 0.4 > 0.38, \end{cases} \tag{3.20}$$

by (3.5) we have $\mathcal{J}_0^{\hat{t}} = \{1, 2, 3, 6\}$.

Step 4: Here, $c = (0.4, 0.5, 0.7, 0.3, 0.5, 0.6)$. Since

$$\begin{aligned} p_0^* &= \arg\min_{j\in\mathcal{J}_0^{\hat{t}}}\{c_j\} = \arg\min\{c_1, c_2, c_3, c_6\} \\ &= \arg\min\{0.4, 0.5, 0.7, 0.6\} = 0.4 = c_1, \end{aligned} \tag{3.21}$$

by (3.8), we find the optimal index as $p_0^* = 1$.

Step 5: Since $p_0^* = 1$, we find the vector t^{*0} by (3.9) as $t^{*0} = (0.38, 0, 0, 0, 0, 0)$. According to Theorem 4, $t^{*0} = (0.38, 0, 0, 0, 0, 0)$ is an optimal solution of subproblem (P_0), i.e., problem (3.19). □

3.3. The SCP-based algorithm for solving problem (1.3)

In Subsection 3.2, we find an optimal solution for each subproblem (P_i), denoted by t^{*i}. Based on these optimal solutions $\{t^{*1}, t^{*2}, \cdots, t^{*m}\}$, we generate the optimal solution of problem (1.3) in this subsection.

For $x^1 = (x_1^1, \cdots, x_n^1), x^2 = (x_1^2, \cdots, x_n^2) \in [0, 1]^n$, let

$$x^1 \vee x^2 = (x_1^1 \vee x_1^2, \cdots, x_n^1 \vee x_n^2). \tag{3.22}$$

Lemma 1. *For arbitrary* $x^1, x^2, \cdots, x^m \in [0, 1]^n$, *we have* $z(x^1 \vee x^2 \vee \cdots \vee x^m) = z(x^1) \vee z(x^2) \vee \cdots \vee z(x^m)$.

Proof. In fact, we have to prove only that $z(x^1 \vee x^2) = z(x^1) \vee z(x^2)$.

$$\begin{aligned} z(x^1 \vee x^2) &= c_1(x_1^1 \vee x_1^2) \vee \cdots \vee c_n(x_n^1 \vee x_n^2) \\ &= (c_1 x_1^1 \vee c_1 x_1^2) \vee \cdots \vee (c_n x_n^1 \vee c_n x_n^2) \\ &= (c_1 x_1^1 \vee \cdots \vee c_n x_n^1) \vee (c_1 x_1^2 \vee \cdots \vee c_n x_n^2) \\ &= z(x^1) \vee z(x^2). \end{aligned} \tag{3.23}$$

□

Lemma 2. *Let* $t \in [0, 1]$ *be an arbitrary real number. Then, we have* $t \in \mathcal{T}^{A,b,d}$ *if and only if* $A \circ t \geq b$ *and* $t \leq \hat{t}$.

Proof. (⇒) This is evident, according to Proposition 1 and the expression of system (1.2).

(⇐) Let us consider, arbitrarily, $i' \in \mathcal{I}$ and $j' \in \mathcal{J}$. Let us recall that $\mathcal{I}_{j'} = \{i \in \mathcal{I} | a_{ij'} > d_i\}$.

If $i' \notin \mathcal{I}_{j'}$, then we have

$$a_{i'j'} \wedge \hat{t}_{j'} \leq a_{i'j'} \leq d_{i'}. \tag{3.24}$$

If $i' \in \mathcal{I}_{j'}$, then by (2.2) we have $I_{j'} \neq \emptyset$ and $\hat{t}_{j'} = \bigwedge_{i \in \mathcal{I}_{j'}} d_i \leq d_{i'}$. Thus,

$$a_{i'j'} \wedge \hat{t}_{j'} \leq \hat{t}_{j'} \leq d_{i'}. \tag{3.25}$$

Due to the arbitrariness of i and j, we have

$$a_{i'j'} \wedge \hat{t}_{j'} \leq d_{i'}, \quad \forall i' \in \mathcal{I}, j' \in \mathcal{J}. \tag{3.26}$$

Hence,

$$a_{i'1} \wedge \hat{t}_1 \vee \cdots \vee a_{i'n} \wedge \hat{t}_n \leq d_{i'}, \quad \forall i' \in \mathcal{I}. \tag{3.27}$$

That is, $A \circ \hat{t} \leq d$. Since $t \leq \hat{t}$, we have $A \circ t \leq A \circ \hat{t} \leq d$. Combining $A \circ t \geq b$, it is immediate that $t \in \mathcal{T}^{A,b,d}$. □

Theorem 5. *Let t^{*i} be an optimal solution of subproblem (P_i) for each $i \in \mathcal{I}$. Then, $t^* = t^{*1} \vee t^{*2} \vee \cdots \vee t^{*m}$ is an optimal solution of problem* (1.3).

Proof. (Feasibility) For arbitrarily given $i \in \mathcal{I}$, we verified in Theorem 4 that $t^{*i} \leq \hat{t}$. Hence,

$$t^* = t^{*1} \vee t^{*2} \vee \cdots \vee t^{*m} \leq \hat{t}. \tag{3.28}$$

Since $t^* = t^{*1} \vee t^{*2} \vee \cdots \vee t^{*m}$, it is obvious that $t^*_j = \bigvee_{k \in \mathcal{I}} t^{*k}_j \geq t^{*i}_j, \forall i \in \mathcal{I}, j \in \mathcal{J}$. Hence, by (3.13),

$$a_{i1} \wedge t^*_1 \vee \cdots \vee a_{in} \wedge t^*_n \geq a_{i1} \wedge t^{*i}_1 \vee \cdots \vee a_{in} \wedge t^{*i}_n = b_i, \quad \forall i \in \mathcal{I}. \tag{3.29}$$

That is, $A \circ t^* \geq b$. Considering $t^* \leq \hat{t}$, it follows from Lemma 2 that $t^* \in \mathcal{T}^{A,b,d}$. That is, t^* is a feasible solution to problem (1.3).

(Optimality) Let $t \in \mathcal{T}^{A,b,d}$ be an arbitrary feasible solution of problem (1.3). By observing system (1.2), it is clear that

$$b_i \leq a_{i1} \wedge t_1 \vee a_{i2} \wedge t_2 \vee \cdots \vee a_{in} \wedge t_n \leq d_i, \quad \forall i \in \mathcal{I}. \tag{3.30}$$

Moreover, by Proposition 1, we have $t \leq \hat{t}$. Hence, t is a feasible solution of subproblem (P_i) for any $i \in \mathcal{I}$. Let us note that t^{*i} is the optimal solution of (P_i). We have

$$z(t) \geq z(t^{*i}), \quad \forall i \in \mathcal{I}. \tag{3.31}$$

Following Lemma 1, we have

$$z(t) \geq z(t^{*1}) \vee z(t^{*2}) \vee \cdots \vee z(t^{*m}) = z(t^{*1} \vee t^{*2} \vee \cdots \vee t^{*m}) = z(t^*). \tag{3.32}$$

As a consequence, t^* is an optimal solution of problem (1.3). □

According to Theorem 5, if we can determine the optimal solutions of all the subproblems $\{P_1, P_2, \cdots, P_m\}$, then the optimal solution of problem (1.3) can be generated by those m optimal solutions. The resolution approach indicated in Theorem 5 is based on single-constraint programming (SCP), i.e., subproblems $\{P_1, P_2, \cdots, P_m\}$. Thus, we call this the SCP-based resolution approach. Moreover, we summarize the resolution approach as the following SCP-based algorithm.

SCP-based Algorithm to obtain an optimal solution of problem (1.3)

Step 1: Construct the vector $\hat{t} = (\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_n)$ following (2.2).

Step 2: Apply the vector $\hat{t}$; check the consistency of system (1.2) following Theorem 1. If $\hat{t} \in \mathcal{T}^{A,b,d}$, then $\mathcal{T}^{A,b,d} \neq \emptyset$ and problem (1.3) is solvable. Continue to the next step. Otherwise, problem (1.3) is unsolvable; stop.

Step 3: Construct m subproblems as (3.4), denoted by $\{(P_1), (P_2), \cdots, (P_m)\}$.

Step 4: For each $i \in \mathcal{I}$, find an optimal solution of the subproblem (P_i) by applying the proposed Algorithm I presented in Subsection 3.2. Suppose the obtained optimal solution of (P_i) is t^{*i}, $\forall i \in \mathcal{I}$. Then, find m optimal solutions as $\{t^{*1}, t^{*2}, \cdots, t^{*m}\}$.

Step 5: Generate the vector $t^* = t^{*1} \vee t^{*2} \vee \cdots \vee t^{*m}$. Then, by Theorem 5, t^* is an optimal solution of problem (1.3).

4. Numerical example

In this section, we provide an illustrative example of our proposed SCP-based algorithm.

Example 2. We consider a P2P network system with 6 terminals. Let us suppose the P2P network system is described by the max-min FRI system as

$$b \leq A \circ t \leq d, \tag{4.1}$$

where

$$A = \begin{pmatrix} 0.3 & 0.7 & 0.6 & 0.8 & 0.6 & 0.7 \\ 0.8 & 0.2 & 0.6 & 0.3 & 0.4 & 0.6 \\ 0.3 & 0.4 & 0.2 & 0.5 & 0.9 & 0.6 \\ 0.9 & 0.6 & 0.8 & 0.2 & 0.3 & 0.4 \\ 0.4 & 0.8 & 0.6 & 0.3 & 0.2 & 0.5 \\ 0.4 & 0.5 & 0.8 & 0.2 & 0.8 & 0.3 \end{pmatrix}, \tag{4.2}$$

$b = (0.45, 0.37, 0.52, 0.38, 0.42, 0.48)$, $d = (0.75, 0.78, 0.85, 0.76, 0.77, 0.89)$, $t = (t_1, t_2, \cdots, t_6)$. We find an optimal solution of the following weighted minimax programming problem:

$$\begin{aligned} \min \quad & z(t) = c_1 t_1 \vee c_2 t_2 \vee \cdots \vee c_6 t_6, \\ \text{s.t.} \quad & b \leq A \circ t \leq d, \end{aligned} \tag{4.3}$$

where $c = (c_1, c_2, \cdots, c_6) = (0.5, 0.6, 0.8, 0.4, 0.6, 0.7)$.

Solution. Step 1: According to $\mathcal{I}_j = \{i \in \mathcal{I} | a_{ij} > d_i\}$, $\mathcal{I}_1 = \{2, 4\}$, $\mathcal{I}_2 = \{5\}$, $\mathcal{I}_3 = \{4\}$, $\mathcal{I}_4 = \{1\}$, $\mathcal{I}_5 = \{3\}$, and $\mathcal{I}_6 = \emptyset$. Based on these index sets, we can calculate the vector $\hat{t}$ by (2.2). After calculation, we have $\hat{t} = (\hat{t}_1, \hat{t}_2, \cdots, \hat{t}_6) = (0.76, 0.77, 0.76, 0.75, 0.85, 1)$.

Step 2: We compute $A \circ \hat{t}$ as follows:

$$\begin{aligned} A \circ \hat{t} &= \begin{pmatrix} 0.3 & 0.7 & 0.6 & 0.8 & 0.6 & 0.7 \\ 0.8 & 0.2 & 0.6 & 0.3 & 0.4 & 0.6 \\ 0.3 & 0.4 & 0.2 & 0.5 & 0.9 & 0.6 \\ 0.9 & 0.6 & 0.8 & 0.2 & 0.3 & 0.4 \\ 0.4 & 0.8 & 0.6 & 0.3 & 0.2 & 0.5 \\ 0.4 & 0.5 & 0.8 & 0.2 & 0.8 & 0.3 \end{pmatrix} \circ (0.76, 0.77, 0.76, 0.75, 0.85, 1) \\ &= (0.75, 0.76, 0.85, 0.76, 0.77, 0.8). \end{aligned} \tag{4.4}$$

We know that $b \leq A \circ \hat{t} \leq d$, i.e., $\hat{t}$ fulfills system (4.1). Hence, system (4.1) is consistent, and problem (4.3) is solvable. We continue to the next step.

Step 3: Following (3.4), we construct the subproblems as follows.

$$(P_1)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.45 \leq 0.3 \wedge t_1 \vee 0.7 \wedge t_2 \vee 0.6 \wedge t_3 \vee 0.8 \wedge t_4 \vee 0.6 \wedge t_5 \vee 0.7 \wedge t_6 \leq 0.75, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

$$(P_2)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.37 \leq 0.8 \wedge t_1 \vee 0.2 \wedge t_2 \vee 0.6 \wedge t_3 \vee 0.3 \wedge t_4 \vee 0.4 \wedge t_5 \vee 0.6 \wedge t_6 \leq 0.78, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

$$(P_3)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.52 \leq 0.3 \wedge t_1 \vee 0.4 \wedge t_2 \vee 0.2 \wedge t_3 \vee 0.5 \wedge t_4 \vee 0.9 \wedge t_5 \vee 0.6 \wedge t_6 \leq 0.85, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

$$(P_4)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.38 \leq 0.9 \wedge t_1 \vee 0.6 \wedge t_2 \vee 0.8 \wedge t_3 \vee 0.2 \wedge t_4 \vee 0.3 \wedge t_5 \vee 0.4 \wedge t_6 \leq 0.76, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

$$(P_5)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.42 \leq 0.4 \wedge t_1 \vee 0.8 \wedge t_2 \vee 0.6 \wedge t_3 \vee 0.3 \wedge t_4 \vee 0.2 \wedge t_5 \vee 0.5 \wedge t_6 \leq 0.77, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

$$(P_6)\ \min\ \ z(t) = 0.5t_1 \vee 0.6t_2 \vee 0.8t_3 \vee 0.4t_4 \vee 0.6t_5 \vee 0.7t_6,$$
$$\text{s.t.}\ \begin{cases} 0.48 \leq 0.4 \wedge t_1 \vee 0.5 \wedge t_2 \vee 0.8 \wedge t_3 \vee 0.2 \wedge t_4 \vee 0.8 \wedge t_5 \vee 0.3 \wedge t_6 \leq 0.89, \\ t \leq (0.76, 0.77, 0.76, 0.75, 0.85, 1). \end{cases}$$

Step 4: In this step, we apply our proposed Algorithm I to find the optimal solutions of the subproblems $\{(P_1), (P_2), \cdots, (P_6)\}$.

According to (3.5), we find the index sets as $\mathcal{J}_1^{\hat{t}} = \{2, 3, 4, 5, 6\}$, $\mathcal{J}_2^{\hat{t}} = \{1, 3, 5, 6\}$, $\mathcal{J}_3^{\hat{t}} = \{5, 6\}$,$\mathcal{J}_4^{\hat{t}} = \{1, 2, 3, 6\}$, $\mathcal{J}_5^{\hat{t}} = \{2, 3, 6\}$, and $\mathcal{J}_1^{\hat{t}} = \{2, 3, 5\}$.

Based on these index sets, we further compute the optimal indices by (3.8) as $p_1^* = 4$, $p_2^* = 1$, $p_3^* = 5$, $p_4^* = 1$, $p_5^* = 2$, and $p_6^* = 2$ or 5.

As a result, we can find the optimal solutions $\{t^{*1}, t^{*2}, \cdots, t^{*6}\}$ following (3.9). Since $p_1^* = 4$, we find the optimal solution to (P_1) as

$$t^{*1} = (0, 0, 0, b_1, 0, 0) = (0, 0, 0, 0.45, 0, 0).$$

Since $p_2^* = 1$, we find the optimal solution to (P_2) as

$$t^{*2} = (b_2, 0, 0, 0, 0, 0) = (0.37, 0, 0, 0, 0, 0).$$

Since $p_3^* = 5$, we find the optimal solution to (P_3) as

$$t^{*3} = (0, 0, 0, 0, b_3, 0) = (0, 0, 0, 0, 0.52, 0).$$

Since $p_4^* = 1$, we find the optimal solution to (P_4) as

$$t^{*4} = (b_4, 0, 0, 0, 0, 0) = (0.38, 0, 0, 0, 0, 0).$$

Since $p_5^* = 2$, we find the optimal solution to (P_5) as

$$t^{*5} = (0, b_5, 0, 0, 0, 0) = (0, 0.42, 0, 0, 0, 0).$$

Since $p_6^* = 2$ or 5, we find the optimal solution to (P_6) as

$$t^{*6} = (0, b_6, 0, 0, 0, 0) = (0, 0.48, 0, 0, 0, 0),$$

or

$$t^{*6} = (0, 0, 0, 0, b_6, 0) = (0, 0, 0, 0, 0.48, 0).$$

Step 5: We generate the vector $t^* = t^{*1} \vee t^{*2} \vee \cdots \vee t^{*6}$. When $t^{*6} = (0, b_6, 0, 0, 0, 0) = (0, 0.48, 0, 0, 0, 0)$, we have

$$\begin{aligned} t^* =& t^{*1} \vee t^{*2} \vee \cdots \vee t^{*6} \\ =&(0, 0, 0, 0.45, 0, 0) \vee (0.37, 0, 0, 0, 0, 0) \vee (0, 0, 0, 0, 0.52, 0) \vee (0.38, 0, 0, 0, 0, 0) \\ &\vee (0, 0.42, 0, 0, 0, 0) \vee (0, 0.48, 0, 0, 0, 0) \\ =&(0.38, 0.48, 0, 0.45, 0.52, 0). \end{aligned}$$

When $t^{*6} = (0, 0, 0, 0, b_6, 0) = (0, 0, 0, 0, 0.48, 0)$, we have

$$\begin{aligned} t^{*\prime} =& t^{*1} \vee t^{*2} \vee \cdots \vee t^{*6} \\ =&(0, 0, 0, 0.45, 0, 0) \vee (0.37, 0, 0, 0, 0, 0) \vee (0, 0, 0, 0, 0.52, 0) \vee (0.38, 0, 0, 0, 0, 0) \\ &\vee (0, 0.42, 0, 0, 0, 0) \vee (0, 0, 0, 0, 0.48, 0) \\ =&(0.38, 0.42, 0, 0.45, 0.52, 0). \end{aligned}$$

According to Theorem 5, both t^* and $t^{*\prime}$ are the optimal solutions of problem (4.3). □

5. Comparing our problem and the proposed method to that in existing research

In this work, we establish a minimax program with max-min FRI constraints. Moreover, we propose the so-called SCP-based algorithm to identify an optimal solution. In the following section, we compare our problem and the proposed resolution algorithm to those presented in several existing works.

(i) The optimization problem investigated in this work has not been studied previously. This result is different from those reported in existing research.

In fact, minimax programming problems subject to addition-min FRIs were studied in [11–17]. In those problems, the optimization objective was a minimax function, but the constraints were the addition-min FRIs; they are different from those in our studied problem. Moreover, [28–34] minimized a linear objective function under the constraints of max-min FRIs, while [18, 19, 35–37] optimized a geometric objective function under the same constraints. Although the constraints are the same as those in our studied problem, the linear or geometric objective function is different from the minimax objective function, which appears in our problem. Consequently, our minimax programming problem with the addition-min FRI constraints is different from those presented in [11–19, 28–37].

(ii) The feasible domain of our optimization model is different from those employed in some relevant published works.

In the minimax optimization problems studied in [11–17], the feasible domain is the solution set to the addition-min FRIs. It has been formally proven that such a feasible domain is a convex set [2]. However, the feasible domain of our problem, i.e., the solution set to the max-min FRIs, should be nonconvex when the minimum is not unique [38]. In addition, a system of addition-min FRIs usually has infinitely many minimal solutions [2], in most cases. However, the number of minimal solutions to the max-min FRIs is always finite. Consequently, the properties of the feasible domain with an addition-min composition are much different from those with a max-min composition.

(iii) The SCP-based algorithm is proposed for our optimization model; this approach is different from the existing resolution methods adopted for the relevant fuzzy relation optimization models in [11–19, 28–37].

In [15], the dichotomy algorithm was proposed for searching for optimal solutions. The subproblem and single-variable approach was also proposed in [11–13, 16, 17]. In addition, [14] developed the so-called optimal-vector-based algorithm for minimax programming with addition-min FRI constraints. However, the branch and bound method [28–34], which is suitable for linear programming with max-min FRI constraints, and the value-matrix-based iterative method [35–37], which is suitable for geometric programming with max-min FRI constraints, are ineffective for our minimax programming problem with max-min FRI constraints. All of these existing methods for the relevant fuzzy relation optimization models are no longer effective for our problem due to their different optimization scenarios and properties of the feasible domains.

6. Conclusions

The P2P network system has been reduced to the max-min FRIs, i.e., system (1.2). To decrease network congestion in the P2P network system, we constructed and investigated a weighted minimax programming problem subject to system (1.2), i.e., problem (1.3). The purpose of this work is to propose an effective algorithm to produce an optimal solution of (1.3). We divided the original problem (1.3) into m subproblems. Each subproblem involves single-constraint programming. Algorithm I was designed to find one of the optimal solutions of each subproblem. The optimal solution can be generated by the optimal solutions from those m subproblems. We further proposed the SCP-based algorithm to find an optimal solution to the original problem (1.3). A numerical example was given to verify the validity of the SCP-based algorithm. Moreover, Example 2 showed that the optimal solutions might not be unique.

In the future, based on the max-min FRIs for modeling the P2P network system, we plan to further consider the stability with respect to some given solutions of the max-min FRIs.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (12271132) and the Guangdong Basic and Applied Basic Research Foundation (2024A1515010532, 2023A1515011093, 2022A1515011460, 2023KQNCX041, 2021ZDJS044, QD202211, PNB2103).

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. J. X. Li, S. J. Yang, *Fuzzy relation equalities about the data transmission mechanism in bittorrent-like peer-to-peer file sharing systems*, In: Proceedings of the 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery, 2012. Available from: https://ieeexplore.ieee.org/abstract/document/6233956.

2. X. P. Yang, H. T. Lin, X. G. Zhou, B. Y. Cao, Addition-min fuzzy relation inequalities with application in BitTorrent-like Peer-to-Peer file sharing system, *Fuzzy Set. Syst.*, **343** (2018), 126–140. https://doi.org/10.1016/j.fss.2017.04.002

3. S. J. Yang, Some results of the fuzzy relation inequalities with addition-min composition, *IEEE T. Fuzzy Syst.*, **26** (2018), 239–245. https://doi.org/10.1109/TFUZZ.2017.2648864

4. B. Y. Cao, X. P. Yang, X. G. Zhou, Properties of fuzzy relation inequalities with addition-min composition, *Adv. Intel. Syst. Comput.*, **646** (2018), 177–185.

5. M. Li, X. P. Wang, Minimal solutions of fuzzy relation inequalities with addition-min composition and their applications, *IEEE T. Fuzzy Syst.*, **31** (2023), 1665–1675.

6. X. P. Yang, G. Z. Zheng, X. G. Zhou, B. Y. Cao, Lexicography minimum solution of fuzzy relation inequalities: Applied to optimal control in P2P file sharing system, *Int. J. Mach. Learn. Cyb.*, **8** (2017), 1555–1563. https://doi.org/10.1007/s13042-016-0527-x

7. X. P. Yang, Random-term-absent addition-min fuzzy relation inequalities and their lexicographic minimum solutions, *Fuzzy Set. Syst.*, **440** (2022), 42–61. https://doi.org/10.1016/j.fss.2021.08.007

8. M. Li, X. P. Wang, Remarks on minimal solutions of fuzzy relation inequalities with addition-min composition, *Fuzzy Set. Syst.*, **410** (2021), 19–26. https://doi.org/10.1016/j.fss.2020.09.014

9. S. M. Guu, Y. K. Wu, A linear programming approach for minimizing a linear function subject to fuzzy relational inequalities with addition-min composition, *IEEE T. Fuzzy Syst.*, **25** (2017), 985–992. https://doi.org/10.1109/TFUZZ.2016.2593496

10. F. F. Guo, J. Shen, A novel smoothing approach for linear objective optimizations subject to fuzzy relation inequalities with addition-min composition, *IEEE T. Fuzzy Syst.*, **29** (2021), 2444–2450. https://doi.org/10.1109/TFUZZ.2020.2991304

11. X. P. Yang, X. G. Zhou, B. Y. Cao, Min-max programming problem subject to addition-min fuzzy relation inequalities, *IEEE T. Fuzzy Syst.*, **24** (2016), 111–119. https://doi.org/10.1109/TFUZZ.2015.2428716

12. Y. Chiu, S. M. Guu, J. Yu, Y. K. Wu, A single-variable method for solving min-max programming problem with addition-min fuzzy relational inequalities, *Fuzzy Optim. Decis. Ma.*, **18** (2019), 433–449. https://doi.org/10.1007/s10700-019-09305-9

13. Y. K. Wu, C. F. Wen, Y. T. Hsu, M. X. Wang, Some results for the minimal optimal solution of min-max programming problem with addition-min fuzzy relational inequalities, *Fuzzy Optim. Decis. Ma.*, **21** (2022), 429–454. https://doi.org/10.1007/s10700-021-09371-y

14. X. P. Yang, Optimal-vector-based algorithm for solving min-max programming subject to addition-min fuzzy relation inequality, *IEEE T. Fuzzy Syst.*, **25** (2017), 1127–1140. https://doi.org/10.1109/TFUZZ.2016.2598367

15. H. Lin, X. Yang, Dichotomy algorithm for solving weighted min-max programming problem with addition-min fuzzy relation inequalities constraint, *Comput. Ind. Eng.*, **146** (2020), 106537. https://doi.org/10.1016/j.cie.2020.106537

16. Y. K. Wu, Y. L. Chiu, S. M. Guu, Generalized min-max programming problems subject to addition-min fuzzy relational inequalities, *Fuzzy Set. Syst.*, **447** (2022), 22–38. https://doi.org/10.1016/j.fss.2022.03.017

17. X. Yang, J. Qiu, H. Guo, X. Yang, Fuzzy relation weighted minimax programming with addition-min composition, *Comput. Ind. Eng.*, **147** (2020), 106644. https://doi.org/10.1016/j.cie.2020.106644

18. X. P. Yang, Linear programming method for solving semi-latticized fuzzy relation geometric programming with max-min composition, *Int. J. Uncertain. Fuzz.*, **23** (2015), 781–804. https://doi.org/10.1142/S0218488515500348

19. X. G. Zhou, X. P. Yang, B. Y. Cao, Posynomial geometric programming problem subject to max-min fuzzy relation equations, *Inform. Sciences*, **328** (2016), 15–25. https://doi.org/10.1016/j.ins.2015.07.058

20. X. B. Yang, X. P. Yang, K. Hayat, A new characterisation of the minimal solution set to max-min fuzzy relation inequalities, *Fuzzy Inf. Eng.*, **9** (2017), 423–435. https://doi.org/10.1016/j.fiae.2017.12.002

21. G. Xiao, T. Zhu, Y. Chen, X. Yang, Linear searching method for solving approximate solution to system of max-min fuzzy relation equations with application in the instructional information resources allocation, *IEEE Access*, **7** (2019), 65019–65028. https://doi.org/10.1109/ACCESS.2019.2912217

22. X. Yang, Evaluation model and approximate solution to inconsistent max-min fuzzy relation inequalities in P2P file sharing system, *Complexity*, **2019** (2019), 6901818. https://doi.org/10.1155/2019/6901818

23. Y. Zhong, G. Xiao, X. Yang, Fuzzy relation lexicographic programming for modelling P2P file sharing system, *Soft Comput.*, **23** (2019), 3605–3614.

24. G. Xiao, K. Hayat, X. Yang, Evaluation and its derived classification in a Server-to-Client architecture based on the fuzzy relation inequality, *Fuzzy Optim. Decis. Ma.*, **22** (2023), 213–245. https://doi.org/10.1007/s10700-022-09390-3

25. Y. B. Ma, X. B. Yang, B. Y. Cao, Fuzzy-relation-based lexicographic minimum solution to the P2P network system, *IEEE Access*, **8** (2020), 195447–195458. https://doi.org/10.1109/ACCESS.2020.3034279

26. Y. Chen, X. Liu, L. Zhang, Interval solution to fuzzy relation inequality with application in P2P educational information resource sharing systems, *IEEE Access*, **9** (2021), 96166–96175. https://doi.org/10.1109/ACCESS.2021.3092745

27. P. Z. Wang, D. Z. Zhang, E. Sanchez, E. S. Lee, Latticized linear programming and fuzzy relation inequalities, *J. Math. Anal. Appl.*, **159** (1991), 72–87. https://doi.org/10.1016/0022-247X(91)90222-L

28. F. F. Guo, L. P. Pang, D. Meng, Z. Q. Xia, An algorithm for solving optimization problems with fuzzy relational inequality constraints, *Inform. Sciences*, **252** (2013), 20–31. https://doi.org/10.1016/j.ins.2011.09.030

29. S. C. Fang, G. Li, Solving fuzzy relation equations with a linear objective function, *Fuzzy Set. Syst.*, **103** (1999), 107–113. https://doi.org/10.1016/S0165-0114(97)00184-X

30. C. W. Chang, B. S. Shieh, Linear optimization problem constrained by fuzzy max-min relation equations, *Inform. Sciences*, **234** (2013), 71–79. https://doi.org/10.1016/j.ins.2011.04.042

31. A. Ghodousian, F. S. Yousefi, Linear optimization problem subjected to fuzzy relational equations and fuzzy constraints, *Iran. J. Fuzzy Syst.*, **20** (2023), 1–20.

32. Z. Matusiewicz, Minimizing and maximizing a linear objective function under a fuzzy max-* relational equation and an inequality constraint, *Kybernetika*, **58** (2022), 320–334.

33. S. M. Guu, Y. K. Wu, Minimizing a linear objective function under a max-t-norm fuzzy relational equation constraint, *Fuzzy Set. Syst.*, **161** (2010), 285–297. https://doi.org/10.1016/j.fss.2009.03.007

34. B. S. Shieh, Minimizing a linear objective function under a fuzzy max-t-norm relation equation constraint, *Inform. Sciences*, **181** (2011), 832–841. https://doi.org/10.1016/j.ins.2010.10.024

35. Y. K. Wu, Optimizing the geometric programming problem with single-term exponents subject to max-min fuzzy relational equation constraints, *Math. Comput. Model.*, **47** (2008), 352–362. https://doi.org/10.1016/j.mcm.2007.04.010

36. G. Singh, D. Pandey, A. Thapar, A posynomial geometric programming restricted to a system of fuzzy relation equations, *Proced. Eng.*, **38** (2012), 3462–3476. https://doi.org/10.1016/j.proeng.2012.06.400

37. B. Hedayatfar, A. A. Molai, Geometric function optimization subject to mixed fuzzy relation inequality constraints, *TWMS J. Appl. Eng. Math.*, **9** (2019), 434–445.
38. P. K. Li, S. C. Fang, On the resolution and optimization of a system of fuzzy relational equations with sup-t composition, *Fuzzy Optim. Decis. Ma.*, **7** (2008), 169–214. https://doi.org/10.1007/s10700-008-9029-y