International Journal of Operations Research Vol. 1, No. 1, 29−36 (2004)

Finding the complete set of minimal solutions for fuzzy relational equations with max-product composition

Yan-Kuen Wu[1,*] and Sy-Ming Guu[2]

[1]Department of Industrial Management, Vanung University, 1 Van-Nung Rd., Chung-Li, Taoyuan, Taiwan, 320, R.O.C.

[2]Department of Business Administration, Yuan Ze University, 135 Far East Rd., Chung-Li, Taoyuan, Taiwan, 320, R.O.C.

Abstract ──It is well known that the solution set of fuzzy relational equations with max-product composition can be determined by the maximum solution and a finite number of minimal solutions. There exists an analytic expression for the maximum solution and it can be yielded easily, but finding the complete set of minimal solutions is not trivial. In this paper we first provide a necessary condition for any minimal solution in terms of the maximum solution. Precisely, each nonzero component of any minimal solution takes the value of corresponding component of the maximum solution. We then propose rules to reduce the problem so that the solution can easily be found. A numerical example is provided to illustrate our procedure.

Keywords ──Fuzzy relational equation, maximum solution, minimal solution

1. INTRODUCTION

Let $A = (a_{ij})$ denote a $m\times n$ nonnegative matrix with $a_{ij} \le 1$ and $b \in R^n$ a nonnegative vector with $b_j \le 1$. The fuzzy relational equations that we consider in this paper are as follows.

$$\begin{aligned} &a_{1j}x_1 \vee a_{2j}x_2 \vee \cdots \vee a_{mj}x_m = b_j \\ &\textit{for all } j = 1, 2, \cdots, n, \end{aligned} \tag{1}$$

where $\vee$ denotes the max operation. We denote the solution set of (1) as $X(A,b)$. Precisely, $X(A, b)$:= $\{\, x \in [0,\ 1]^m \mid x \circ A = b \}$, where the operation $\circ$ denotes the max-product composition.

The original study of fuzzy relational equations with max-product composition goes back to Pedrycz (1985). Recent study of (1) can be found in Bourke and Fisher (1998), Loetamonphong and Fang (1999). Furthermore, the monograph by Di Nola, Sessa, Pedrycz and Sanchez (1989) contains a thorough discussion of this class of equations. The notion of fuzzy relational equations with max-min composition was first proposed and studied by Sanchez (1976) (see Czogala, Drewniak and Pedrycz (1982), Higashi and Klir (1984) as well.) Applications of fuzzy relational equations can be found in Sanchez (1977).

Related topics are optimization problems with objective functions subjected to the fuzzy relational equation constraints. Fang and Li (1999) was the first paper to consider the fuzzy relational equations with a linear objective function, where the algebraic operations employed in fuzzy relational equations are the max-min composition. Wu, Guu and Liu (2002) improved Fang and Li's method by providing an upper bound for the branch and bound procedure. Since $X(A,b)$ is non-convex, Lu and Fang (2001) proposed a genetic algorithm to solve the problems. Lee and Guu (2003) proposed a fuzzy relational optimization model for the streaming media provider seeking a minimum cost while fulfilled the requirements assumed by a three-tier framework. Wang (1995) was the first paper to explore the same optimization problem yet with multiple linear objective functions. Recently, Loetamonphong, Fang and Young (2002) have studied the multi-objective optimization problem with nonlinear objective functions. A genetic algorithm was employed to find the Pareto optimal solutions.

On the other hand, Loetamonphong and Fang (2001) was the first paper to consider similar optimization models where the algebraic operations in fuzzy relational equation constraints are the max-product composition. Motivated by the network reliability problems, Guu and Wu (2002) studied such models and provided a necessary condition for an optimal solution to hold. Based on this necessary condition, efficient procedures were proposed to find an optimal solution.

For convenience, we let I={1, 2, …, n} and J={1, 2, …, n}. If we define $x^1 \le x^2$ if and only if $x_i^1 \le x_i^2$ for $i \in I$, then the operator $\le$ forms a partial order relation on $X(A,b)$. An $\bar{x} \in X(A,b)$ is the maximum solution if $x \le \bar{x}$ for all $x \in X(A,b)$. On the other hand, an $\underline{x} \in X(A,b)$ is a minimal solution if $\forall\, x \in X(A,b)$, $x \le \underline{x}$ implies that $x = \underline{x}$. It is well known (see, for instance, Bourke and Fisher (1998)) that when $X(A,b)$ is nonempty, the complete solution set $X(A,b)$ can be determined by a unique maximum solution and a finite number of minimal solutions. Moreover, the potential maximum solution can be obtained by the following operation:

[*]Corresponding author's email: ykw@msa.vnu.edu.tw

$$\bar{x} = A \Diamond b = [\min_{j \in J}(a_{ij} \Diamond b_j)]_{\ i \in I}, \quad (2)$$

Where

$$a_{ij} \Diamond b_j := \begin{cases} 1 & \text{if} \quad a_{ij} \le b_j; \\ \frac{b_j}{a_{ij}} & \text{if} \quad a_{ij} > b_j. \end{cases}$$

Note that $X(A,b)$ is nonempty if and only if the vector $A \Diamond b$ is a solution of (1). We assume in this paper that $X(A,b)$ is nonempty.

Although the maximum solution can be easily obtained, finding the complete set of minimal solutions is not trivial. Bourke and Fisher (1998) were the first to propose algorithms to find the complete set of minimal solutions. Since the solution set $X(A,b)$ is non-convex, it turns out that the total number of minimal solutions has a combinatorial nature in terms of the problem size. To overcome the combinatorial nature, Leotamonphong and Fang (1999) explored the special structure of the solution set to reduce the problem size. Leotamonphong and Fang reported that their procedure is more effective than those in Bourke and Fisher.

Our contribution in this paper is for computing the set of minimal solutions. Let x^* be a minimal solution of (1) and $\bar{x}$ be its maximum solution. We prove that x_i^* is either 0 or $\bar{x}_i$ for each $i = 1, 2, \cdots, m$, that is, each nonzero component of any minimal solution of (1) takes the value of corresponding component of the maximum solution. Based on this necessary condition, we then improve Leotamonphong and Fang's method by proposing four rules to reduce the problem size (let us note that the rule employed in Leotamonphong and Fang (1999) is the rule 1 here.) An example is proposed to illustrate the difference between Leotamonphong and Fang and our procedure.

This paper is organized as follows. In Section 2, we prove a necessary condition for a minimal solution to hold and provide four rules to reduce the problem. Section 3 contains an example to illustrate the procedure in our algorithm. Brief conclusion is given in Section 4.

2. NECESSARY CONDITION AND RULES FOR REDUCING THE PROBLEM

In this section we investigate the properties of the solution set of equation (1).

Lemma 1: Assume $x \in X(A,b) \ne \phi$. If there exists an index j with $b_j = 0$ and $a_{ij} > 0$ for all $i \in I$ in problem (1), then equation (1) has only zero solution.

Proof: Due to $b_j = 0$, the j equation becomes $\max_{i \in I}\{x_i a_{ij}\} = 0$. Since $0 \le x_i \le 1$ and $0 < a_{ij} \le 1$, this leads $x_i = 0$ for all $i \in I$.

Lemma 1 illustrates that if for some j there exists $b_j = 0$ and $a_{ij} > 0$ for all $i \in I$ in (1), then the solution set is trivial.

In the following, we shall assume that $b_j > 0$ for all $j \in J$.

Lemma 2: If there exists an index j with $a_{ij} < b_j$ for each $i \in I$ in (1), then $X(A,b) = \phi$.

Proof: Obviously.

If $X(A,b) \ne \phi$ and $x \in X(A,b)$, then there exists at least one index $i_0 \in I$ such that $x_{i_0} a_{i_0 j} = b_j$ for each $j \in J$.

Definition 1: For any solution x in $X(A,b)$, we call the component x_{i_0} a binding variable if there exists $j \in J$ such that $x_{i_0} a_{i_0 j} = b_j$.

Due to the assumption of $b_j > 0$ for all $j \in J$, if $a_{i_0 j} = 0$ in the jth equation, then variable x_{i_0} can't be binding in jth equation. On the other hand, if x_{i_0} is binding in jth equation, then $a_{i_0 j} > 0$.

Lemma 3: If there exists $a_{ij} < b_j$ for all $j \in J$, then $x_i = 0$ for any minimal solution x.

Proof: Since $0 \le x_i \le 1$, when $a_{ij} < b_j$ for all $j \in J$, this leads $x_i a_{ij} < b_j$, $\forall j \in J$ in (1). This result shows that variable x_i can't satisfy any equation in (1). Hence, setting $x_i = 0$ does not affect the solution set.

Lemma 3 also reveals that if $a_{ij} < b_j$ for any $j \in J$, then x_i can't be binding in jth equation. On the other hand, a necessary condition for x_i to be binding in jth equation is $a_{ij} \ge b_j$.

Lemma 4: Let $x \in X(A,b)$. If x_i is a binding variable, then $x_i = \min\{\frac{b_j}{a_{ij}} \mid a_{ij} \ne 0 \text{ and } j \in J\}$.

Proof: For a solution $x \in X(A,b)$, we have $\max_{i \in I}\{x_i a_{ij}\} = b_j$, $\forall j \in J$. For the variable x_i, we have $x_i a_{ij} \le b_j$, $\forall j \in J$. Since x_i is a binding variable, we have $x_i a_{ij} = b_j$, hence $x_i = b_j / a_{ij}$ for some j and $x_i = \min\{\frac{b_j}{a_{ij}} \mid a_{ij} \ne 0 \text{ and } j \in J\}$.

In fact, Lemma 4 shows that if x_i is a binding variable for any feasible solution, then the solution value of x_i is unique and $x_i > 0$. Moreover, employing above mentioned Lemmas and maximum solution (2), we are ready to provide a necessary condition of problem for a minimal solution.

Theorem 1: Let $\bar{x}$ be the maximum solution and x be a solution of (1). If x_i is a binding variable, then $x_i = \bar{x}_i$.

Proof: Since $\bar{x}$ is the maximum solution, we have

$\bar{x} = A \Diamond b = [\min_{j\in J}(a_{ij} \Diamond b_j)]_{i\in I}$,

where

$$a_{ij} \Diamond b_j := \begin{cases} 1 & \text{if} \quad a_{ij} \le b_j; \\ \frac{b_j}{a_{ij}} & \text{if} \quad a_{ij} > b_j. \end{cases}$$

For each $i \in I$, we can rewrite $\bar{x}_i = \min_{j\in J}\{ 1, \frac{b_j}{a_{ij}} \}$. Since x_i is a binding variable, we have $a_{ij} \ge b_j$. It follows that $x_i = \min\{ \frac{b_j}{a_{ij}} \mid a_{ij} \ne 0 \text{ and } j \in J\}$. By Lemma 4, we have $x_i = \bar{x}_i$.

Theorem 2: If x^* is a minimal solution, then we have either $x_i^* = 0$ or $x_i^* = \bar{x}_i$ for each $i \in I$.

Proof: Each variable in a minimal solution is either non-binding or binding. Suppose that x_i^* is not a binding variable and $x_i^* > 0$. Then there exists a solution $x^! < x^*$ and $x_i^! = 0 < x_i^*$ since that we can choose $x_k^! = x_k^*$ for all $k \in I$ and $k \ne i$. This implies that x^* is not a minimal solution. Hence, a non-binding x_i^* implies $x_i^* = 0$. On the other hand, if x_i^* is a binding variable, then by Theorem 1, we have $x_i^* = \bar{x}_i$.

In fact, Theorem 2 reveals the necessary condition of a minimal solution. It describes that for any minimal solution x^*, if x_i^* is not a binding variable, then x_i^* can be assigned to zero. On the other hand, if x_i^* is a binding variable, then x_i^* equals to $\bar{x}_i$. It turns out that the maximum solution $\bar{x}$ can provide useful information in searching for the minimal solutions. To do so, we define the following index sets.

$J_i := \{ j \in J \mid \bar{x}_i a_{ij} = b_j \}$, $\forall\, i \in I$ and $I_j := \{ i \in I \mid \bar{x}_i a_{ij} = b_j \}$, $\forall\, j \in J$.

The index set J_i indicates those equations satisfied by $\bar{x}_i$. For any minimal solution x, $|J_i|$ is the number of equations in which x_i may become binding; while the index set I_j indicates those possible components (decision variables) of x to be binding in the jth equation. Therefore, we have $|I_1|$ ways to select a binding variable in the first equation and $|I_2|$ in the second equation, etc. In total, we have $|I_1| \times |I_2| \times \cdots \times |I_n|$ ways for problem (1). This quantity can be employed as the problem complexity.

Since for any minimal solution x each component of x is either 0 or $\bar{x}_i$, to compute a minimal solution, we only need to determine which component is zero and which component is not zero (and hence assign the corresponding component of $\bar{x}$ to it.) Theorem 2 implies that the indices of those nonzero components of any minimal solution x^* are contained in the index set $\cup_{i\in I} J_i$. Hence, we define a matrix $M = [m_{ij}]$ with $i \in I$ and $j \in J$ by

$$m_{ij} = \begin{cases} 1 & \text{if } j \in J_i; \\ 0 & \text{otherwise.} \end{cases}$$

Note that the entry $m_{ij} = 1$ in M corresponds to a possible selection of the ith component of some minimal solutions to be binding in the jth equation. Since each solution must satisfy all equations, thus, the procedure of finding minimal solutions can be transformed into the selection of one entry with value 1 in each column of matrix *M*. Moreover, the selection should use the least number of binding variables (hence the entries with value 1 in matrix *M*) to satisfy all equations.

Rule 1: If there exists a singleton $I_j = \{i\}$ for some $j \in J$ in matrix M, we assign $\bar{x}_i$ to the ith component of any minimal solutions. (This is the rule employed by Loetamonphong and Fang (1999).) The index set $I_j = \{i\}$ identifies the jth equation that can be satisfied only by variable x_i in problem. This leads that the ith component of any minimal solutions (hence, the variable x_i) must be binding in the jth equation. Thanks to $x_i = \bar{x}_i$ by Theorem 2, we can delete the jth column of M with $j \in J_i$ from further consideration. The corresponding row of x_i in M can be deleted as well. Note that, after deletion during the process of finding minimal solutions, we let $\hat{J}$ represent the index set of the remaining columns in the reduced matrix. Set $\hat{I}$ denotes the remaining variables which associated with the rows of the reduced matrix.

Rule 2: In the process of finding minimal solutions, for some $x_k \in \hat{I}$ if there exists $J_k = \phi$ in the reduced matrix, then we can assign the value 0 to the kth component of minimal solutions. Since $J_k = \phi$, this implies that the kth component of minimal solutions is not a binding variable to the remaining equations. Hence, we can assign 0 to the kth component of minimal solutions. Moreover, the row of the reduced matrix that associated with x_k can be deleted from further consideration. And x_k can be eliminated from set $\hat{I}$.

Rule 3: In the process of finding minimal solutions, for some $x_s \in \hat{I}$ if there exists $J_s = \hat{J}$ in the reduced matrix, then we assign $x_s^* = \bar{x}_s$ as a binding variable to yield a minimal solution. It implies that all equations have been satisfied by the assigned binding variables. Hence, it yields a minimal solution x^* with components $x_i^* = \bar{x}_i$

if x_i^* is a binding variable (surely includes $x_s^* = \bar{x}_s$) and $x_i^* = 0$ if x_i^* is not a binding variable. Moreover, the row of the reduced matrix that associated with x_s can be deleted from further consideration. And x_s can be eliminated from set $\hat{I}$.

Rule 4: In the process of finding minimal solutions, for some $p \in \hat{J}$ if there exists $I_p = \phi$ in the reduced matrix, then we can't find any minimal solutions for the remaining equations. During the process of finding minimal solutions, it may happen the situation that we have some assigned binding variables, say $x_{i_1}^*, \cdots, x_{i_k}^*$, but some remaining equations, assume the *p*th equation is included, still unsatisfy. If there exists $I_p = \phi$, $p \in \hat{J}$, it implies that the *p*th equation can't be satisfied by the remaining variables. Hence, we can't find any minimal solution with the type of $(\bar{x}_{i_1}^*, \cdots, \bar{x}_{i_k}^*, 0, \cdots, 0)$. This situation indicates that the process of finding minimal solutions should turn to next iteration or stop (to be identified shortly in Step 9).

Employing these rules on matrix *M*, we are ready to present the procedure to find the complete set of minimal solutions for equation (1). The intuition behind the procedure is to apply rules 1-4 to fix as many as possible the binding variables such that some components of minimal solutions can be determined. Then the problem size is reduced by eliminating the corresponding rows and columns associated with binding variable from matrix *M*. When the problem can't be reduced any more, we arrange the rows of the reduced matrix to the decreasing order of $|J_i|$. The arranged matrix shows that the first variable can satisfy the largest number of equations for the reduced problem. So we select it as the binding variable to find minimal solutions. Rules 1-4 apply again to fix some variables to fill in the missing equations. We iterate the process step-by-step until the Stop flag is switched on (i.e. the rule 4 is satisfied) or all of the variables have been considered. Now the procedure for obtaining the complete set of minimal solutions can be summarized as follows.

Step 1 Compute the maximum solution $\bar{x}$ by (2).

Step 2 Check the consistency by verifying whether $\bar{x} \circ A = b$. If inconsistent, then Stop.

Step 3 Compute index sets I_j for all $j \in J$ and J_i for all $i \in I$.

Step 4 Obtain the matrix M.

Step 5 Apply rule 1, rule 2 and rule 3 to fix as many as possible values of components of minimal solutions. If rule 4 is satisfied, then Stop the procedure.

Step 6 Arrange the rows of matrix M according to the decreasing order of $|J_i(M)|$ and denote the arranged matrix by $\bar{M}$. (Now the cardinality of $J_i(M)$ represents the number of columns with value 1 in the ith row of matrix M.) Then record the variables associated with the rows of matrix $\bar{M}$ in order and denote it by set $\bar{I}$.

Step 7 Select the first entry from set $\bar{I}$ as a binding variable, say x_i, then eliminate the corresponding rows and columns associated with binding variable x_i from matrix $\bar{M}$. Denote the reduced matrix by $\bar{M}_i$ and arrange the rows in decreasing order of $|J_i(\bar{M}_i)|$. By the reduced matrix $\bar{M}_i$, we obtain set $\hat{I}$ and index set $\hat{J}$.

Step 8 Apply rules 1-3 to fix values of components of minimal solutions to the reduced matrix $\bar{M}_i$ until no any rules can be satisfied again.

Step 9 We have two cases to be considered. Case 1: if there don't leave any remaining rows or columns in $\bar{M}_i$ (i.e. set $\hat{I} = \phi$ or $\hat{J} = \phi$), then eliminate the first row from matrix $\bar{M}$ and first entry from set $\bar{I}$. Check the rule 4 to matrix $\bar{M}$, if the rule 4 satisfies then go to Step 10, otherwise set the assigned binding variable $x_i = 0$ and go to Step 7 until set $\bar{I}$ equals to empty set. Case 2: if there exists some remaining rows in $\bar{M}_i$ (i.e. set $\hat{I} \neq \phi$ and $\hat{J} \neq \phi$), then we arrange the remaining rows of $\bar{M}_i$ according to the decreasing order of $|J_i(\bar{M}_i)|$. Record the variables associated with the rows of matrix $\bar{M}_i$ in order and denote it by set new-$\bar{I}$. Look upon set new-$\bar{I}$ as set $\bar{I}$ as if we were facing a reduced-problem and go to Step 7.

Step 10 Delete the nonminimal solution and print the complete set of minimal solutions.

3. AN EXAMPLE

In this section, an example is applied to demonstrate how our procedure employs a matrix and rules 1-4 to determine the complete set of minimal solutions efficiently. This example also illustrates that Leotamonphong and Fang's method will generate nonminimal solutions.

Example: Consider the following problem of fuzzy relational equations with max-product composition.

$x \circ A = b$

where

$$x = [\, x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \,],$$

$$A = \begin{bmatrix} 0.5 & 0.6 & 0.12 & 0.45 & 0.4 & 0.45 & 0.3 \\ 0.4 & 0.35 & 0.3 & 0.7 & 0.35 & 0.4 & 0.4 \\ 0.5 & 0.96 & 0.42 & 0.5 & 0.64 & 0.32 & 0.2 \\ 0.5 & 0.8 & 0.35 & 0.3 & 0.28 & 0.6 & 0.1 \\ 0.8 & 0.5 & 0.25 & 0.98 & 0.64 & 0.72 & 0.6 \\ 0.2 & 0.8 & 0.35 & 0.36 & 0.42 & 0.6 & 0.5 \end{bmatrix},$$

$$b = [\, 0.4 \quad 0.48 \quad 0.21 \quad 0.49 \quad 0.32 \quad 0.36 \quad 0.3 \,].$$

Step 1. Compute the maximum solution

$\bar{x} = [\, 0.8 \quad 0.7 \quad 0.5 \quad 0.6 \quad 0.5 \quad 0.6 \,]$.

Step 2. Direct computing shows that $\bar{x} \circ A = b$ holds. Hence, the problem is solvable and $X(A, b) \neq \phi$.

Step 3. Compute index sets I_j for all $j \in J$ and J_i

for all $i \in I$. They are
$I_1 = \{1, 5\}$, $I_2 = \{1, 3, 4, 6\}$, $I_3 = \{2, 3, 4, 6\}$,
$I_4 = \{2, 5\}$, $I_5 = \{1, 3, 5\}$, $I_6 = \{1, 4, 5, 6\}$,
$I_7 = \{5, 6\}$; $J_1 = \{1, 2, 5, 6\}$, $J_2 = \{3, 4\}$,
$J_3 = \{2, 3, 5\}$, $J_4 = \{2, 3, 6\}$, $J_5 = \{1, 4, 5, 6, 7\}$,
$J_6 = \{2, 3, 6, 7\}$.

Note that the problem complexity is 1,536 ($= 2\times4\times4\times2\times3\times4\times2$.)

Step 4. Obtain the matrix *M*. We have

$$\begin{array}{c} \text{equation} \rightarrow \\ \\ \\ M = \\ \\ \\ \\ \end{array} \begin{array}{c} \\ (x_1) \\ (x_2) \\ (x_3) \\ (x_4) \\ (x_5) \\ (x_6) \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \end{array}.$$

Step 5. Apply rule 1, rule 2 and rule 3 to fix as many as possible values of components of minimal solutions. If rule 4 is satisfied, then Stop the procedure.

In this stage, no rules can be applied to reduce or Stop the original problem.

Step 6. Arrange the rows of matrix *M* according to the decreasing order of $|J_i(M)|$ and denote the arranged matrix by $\bar{M}$.

We compute $|J_i(M)|$ and yield the order

$$|J_5(M)| = 5 > |J_1(M)| = |J_6(M)| = 4 > |J_3(M)| = |J_4(M)|$$
$$= 3 > |J_2(M)| = 2.$$

Then we obtain the arranged matrix $\bar{M}$ as follows.

$$\begin{array}{c} \text{equation} \rightarrow \\ \\ \\ \bar{M} = \\ \\ \\ \\ \end{array} \begin{array}{c} \\ (x_5) \\ (x_1) \\ (x_6) \\ (x_3) \\ (x_4) \\ (x_2) \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \end{array}.$$

Recording the variables associated with the rows of matrix $\bar{M}$ in order, we have index set $\bar{I} = \{x_5, x_1, x_6, x_3, x_4, x_2\}$.

Step 7. Select the first entry from set $\bar{I}$ as a binding variable, say x_i, then eliminate the corresponding rows and columns associated with x_i from matrix $\bar{M}$.

Now there are six variables in set $\bar{I}$. Each of these variables can be binding. However, in order to use the least number of binding variables satisfies the remaining equations to find minimal solutions, we select first variable x_5 as a binding variable. By Theorem 2, we assign $x_5 = \bar{x}_5 = 0.5$.

Note that x_5 is binding in equations 1,4,5,6 and 7 (or columns 1,4,5,6,7 of $\bar{M}$). Hence, these columns and first row can be deleted from consideration. After deletion and arrangement by order of $|J_3(\bar{M}_5)| = |J_4(\bar{M}_5)| = |J_6(\bar{M}_5)| = 2 > |J_1(\bar{M}_5)| = |J_2(\bar{M}_5)| = 1$, the reduced matrix (denoted by $\bar{M}_5$) becomes

$$\begin{array}{c} \text{equation} \rightarrow \\ \\ \\ \bar{M}_5 = \\ \\ \\ \end{array} \begin{array}{c} \\ (x_3) \\ (x_4) \\ (x_6) \\ (x_1) \\ (x_2) \end{array} \begin{array}{c} \begin{array}{cc} 2 & 3 \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array}.$$

By the reduced matrix $\bar{M}_5$, we obtain sets $\hat{I} = \{x_3, x_4, x_6, x_1, x_2\}$ and $\hat{J} = \{2, 3\}$.

Step 8. Apply rules 1-3 to fix values of components of minimal solutions to matrix $\bar{M}_5$ until no any rules can be satisfied again.

For $\bar{M}_5$, it is the situation as if we were facing two equations with 5 variables. We can compute the index sets with respect to $\bar{M}_5$ to yield $J_3(\bar{M}_5) = J_4(\bar{M}_5) = J_6(\bar{M}_5) = \{2, 3\}$. It just satisfies the remaining equations in index set $\hat{J}$. By rule 3, this implies that we can yield three minimal solutions. And they are
$\underline{x}^1 = (0, 0, 0.5, 0, 0.5, 0)$,
$\underline{x}^2 = (0, 0, 0, 0.6, 0.5, 0)$ and
$\underline{x}^3 = (0, 0, 0, 0, 0.5, 0.6)$.

Note that the variables x_3, x_4 and x_6 in $\bar{M}_5$ have been considered as binding variables respectively. Hence, we delete the corresponding rows of variables x_3, x_4 and x_6 from $\bar{M}_5$. The reduced matrix becomes

$$\begin{array}{c} \text{equation} \rightarrow \\ \bar{M}_5 = \\ \\ \end{array} \begin{array}{c} \\ (x_1) \\ (x_2) \end{array} \begin{array}{c} \begin{array}{cc} 2 & 3 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{array}.$$

For the current matrix, we have set $\hat{I} = \{x_1, x_2\}$ and assign $x_1 = \bar{x}_1 = 0.8$ and $x_2 = \bar{x}_2 = 0.7$ as binding variables by rule 1. The remaining equations have been satisfied to yield another minimal solution
$\underline{x}^4 = (0.8, 0.7, 0, 0, 0.5, 0)$.

We delete the corresponding rows of x_1 and x_2 in $\bar{M}_5$ by rule 1. Now no remaining rows or columns in $\bar{M}_5$ are left to be considered and set $\hat{I} = \phi$.

Step 9. Now the reduced matrix $\bar{M}_5$ with set $\hat{I} = \phi$, hence, the case 1 should be considered. We eliminate the first row from $\overline{M}$ and it becomes

$$\begin{array}{r} \text{equation} \rightarrow \\ \\ (x_1) \\ (x_6) \\ \overline{M} = (x_3) \\ (x_4) \\ (x_2) \end{array} \begin{array}{c} 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7 \\ \left[\begin{array}{ccccccc} 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{array}\right] \end{array}.$$

We eliminate the first entry x_5 from set $\overline{I}$ and it becomes $\overline{I} = \{x_1,\ x_6,\ x_3,\ x_4,\ x_2\}$. Now the rule 4 is unsatisfied to matrix $\overline{M}$, we set $x_5 = 0$ to the follow-up process and go to Step 7.

Step 7. Now finding minimal solutions, we select first entry x_1 from set $\overline{I}$ as a binding variable and with x_5 a nonbinding variable. We assign $x_1 = \overline{x}_1 = 0.8$ and $x_5 = 0$.

Note that variable x_1 is binding in the first, second, 5th and 6th equations to $\overline{M}$ now. Hence, these corresponding columns and row can be deleted. After deletion and arrangement by order of $|J_2(\overline{M}_1)| = |J_6(\overline{M}_1)| = 2 > |J_3(\overline{M}_1)| = |J_4(\overline{M}_1)| = 1$, the reduced matrix (denoted by $\overline{M}_1$) becomes

$$\begin{array}{r} \text{equation} \rightarrow \\ \\ (x_2) \\ \overline{M}_1 = (x_6) \\ (x_3) \\ (x_4) \end{array} \begin{array}{c} 3\ \ 4\ \ 7 \\ \left[\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{array}\right] \end{array}.$$

By the reduced matrix $\overline{M}_1$, we obtain sets $\hat{I} = \{x_2,\ x_6,\ x_3,\ x_4\}$ and $\hat{J} = \{3,\ 4,\ 7\}$.

Step 8. Apply rules 1-3 to fix values of components of minimal solutions to matrix $\overline{M}_1$ until no any rules can be satisfied again.

For matrix $\overline{M}_1$, it is as if we were facing three equations with 4 variables. We compute the index set $I_4(\overline{M}_1) = \{2\}$ and $I_7(\overline{M}_1) = \{6\}$. By rule 1, we shall set $x_2 = \overline{x}_2 = 0.7$ and $x_6 = \overline{x}_6 = 0.6$. Since all equations are satisfied by x_2 and x_6, the remaining variables x_3 and x_4 are assigned to be zero by rule 2. We obtain the fifth minimal solution as follows:

$$\underline{x}^5 = (\ 0.8,\ \ 0.7,\ \ 0,\ \ 0,\ \ 0,\ \ 0.6\ ).$$

Now no remaining rows or columns in $\overline{M}_1$ are left to be considered and set $\hat{I} = \phi$.

Step 9. Now the reduced matrix $\overline{M}_1$ with set $\hat{I} = \phi$, hence, the case 1 should be considered. We eliminate the first row from $\overline{M}$ and it becomes

$$\begin{array}{r} \text{equation} \rightarrow \\ (x_6) \\ (x_3) \\ \overline{M} = \\ (x_4) \\ (x_2) \end{array} \begin{array}{c} 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7 \\ \left[\begin{array}{ccccccc} 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{array}\right] \end{array}.$$

We eliminate the first element x_1 from set $\overline{I}$ and it becomes $\overline{I} = \{x_6,\ x_3,\ x_4,\ x_2\}$.

For the current matrix $\overline{M}$, we discover set $I_1(\overline{M}) = \phi$. The rule 4 is satisfied, it implies that the first equation can't be satisfied again by remaining variables. In this situation, no more potential minimal solutions can be generated and Stop flag is switched on. Hence, the complete set of minimal solutions has been found. Go to Step 10.

Step 10. Delete the nonminimal solutions and print the complete set of minimal solutions.

For this Example, our procedure yields five minimal solutions and doesn't generate any nonminimal solutions. They are

$\underline{x}^1 = (\ 0,\ \ 0,\ \ \overline{x}_3,\ \ 0,\ \ \overline{x}_5,\ \ 0\ )$,

$\underline{x}^2 = (\ 0,\ \ 0,\ \ 0, \overline{x}_4,\ \ \overline{x}_5,\ \ 0\ )$,

$\underline{x}^3 = (\ 0,\ \ 0,\ \ 0,\ \ 0,\ \ \overline{x}_5,\ \ \overline{x}_6\ )$,

$\underline{x}^4 = (\overline{x}_1,\ \ \overline{x}_2,\ \ 0,\ \ 0,\ \ \overline{x}_5,\ \ 0\ )$ and

$\underline{x}^5 = (\overline{x}_1,\ \ \overline{x}_2,\ \ 0,\ \ 0,\ \ 0,\ \ \overline{x}_6)$.

Now, we apply the method proposed by Leotamonphong and Fang in 1999 to solve Example again. The step-by-step results generated by their method, we find that their algorithm reports 11 solutions as following Table 1. Among them, 5 solutions are the minimal solutions (which we have attached a * in the “minimal solution” column.)

Numerical Experiment: In Table 2, we have compared the performance of our procedure and Leotamonphong and Fang's method. Here we use the same test examples. The current experiment was programmed by Visual Basic 6.0 on a Pentium III PC with 1000 MHZ and 256-MB RAM. Note that the largest test problem for Table 2 is as following matrix A with size 15×20 and vector b with size 1×20. All of the other test problems derive mainly from the largest test problem. For example, we delete the last three rows and five columns from the largest matrix A and delete last five entries from vector b, then the test problem No. 9 contains matrix A with size of problem $m = 12$ and $n = 15$ and the corresponding vector b with size 1×15. Moreover, the other test problems with different size of problem can be generated by the same pattern. In general, the testing problems of numerical experiment may randomly generate, but the max-product fuzzy relational equations with large number of equations, say $n \geq 20$, seem very difficult to be consistent. Although Table 2 shows that our procedure determines fewer solutions than Leotamonphong and Fang's method in each test problem, it still obtains a few nonminimal solutions.

Table 1. Results of Leotamonphong and Fang's method for Example

Minimal solution		Remark
$\underline{x}^1 = (\bar{x}_1,\ \bar{x}_2,\ 0,\ 0,\ \bar{x}_5,\ 0)$	*	
$\underline{x}^2 = (\bar{x}_1,\ 0,\ \bar{x}_3,\ 0,\ \bar{x}_5,\ 0)$		nonminimal solution
$\underline{x}^3 = (\bar{x}_1,\ 0,\ 0,\ \bar{x}_4,\ \bar{x}_5,\ 0)$		nonminimal solution
$\underline{x}^4 = (\bar{x}_1,\ 0,\ 0,\ 0,\ \bar{x}_5,\ \bar{x}_6)$		nonminimal solution
$\underline{x}^5 = (0,\ 0,\ \bar{x}_3,\ 0,\ \bar{x}_5,\ 0)$	*	
$\underline{x}^6 = (0,\ 0,\ 0,\ \bar{x}_4,\ \bar{x}_5,\ 0)$	*	
$\underline{x}^7 = (0,\ 0,\ 0,\ 0,\ \bar{x}_5,\ \bar{x}_6)$	*	
$\underline{x}^8 = (\bar{x}_1,\ \bar{x}_2,\ 0,\ 0,\ 0,\ \bar{x}_6)$	*	
$\underline{x}^9 = (\bar{x}_1,\ \bar{x}_2,\ \bar{x}_3,\ 0,\ 0,\ \bar{x}_6)$		nonminimal solution
$\underline{x}^{10} = (\bar{x}_1,\ \bar{x}_2,\ 0,\ \bar{x}_4,\ 0,\ \bar{x}_6)$		nonminimal solution
$\underline{x}^{11} = (\bar{x}_1,\ \bar{x}_2,\ 0,\ 0,\ 0,\ \bar{x}_6)$		duplicated

Table 2. Performance of our procedure and Leotamonphong and Fang's method

No.	Size of problem (m, n)	problem complexity	Number of minimal solutions	Our procedure		Leotamonphong and Fang's method	
				No. of solutions	cpu time (sec.)	No. of solutions	cpu time (sec.)
1	(15, 20)	5.12E+14	93	113	0.2734	3,152	17.8906
2	(15, 18)	1.71E+13	85	91	0.2578	2,787	14.2031
3	(15, 16)	7.11E+11	90	96	0.2344	1,510	4.9063
4	(15, 15)	1.78E+11	100	110	0.2734	1,486	4.7813
5	(15, 12)	2.22E+09	84	95	0.1875	544	0.9375
6	(12, 20)	3.66E+12	16	17	0.0313	41	0.0469
7	(12, 18)	1.46E+11	16	17	0.0234	41	0.0547
8	(12, 16)	2.09E+10	27	28	0.0547	370	0.4531
9	(12, 15)	6.97E+09	30	33	0.0547	362	0.4531
10	(12, 12)	2.58E+08	34	38	0.0469	185	0.1797
11	(10, 20)	1.88E+10	6	7	0.0234	17	0.0313
12	(10, 18)	1.18E+09	6	7	0.0234	17	0.0234
13	(10, 16)	1.96E+08	10	10	0.0234	128	0.1016
14	(10, 15)	9.80E+07	12	13	0.0156	123	0.0859
15	(10, 12)	1.22E+07	21	22	0.0313	97	0.0703

The largest test problem of fuzzy relational equations for Table 2 with max-product composition is described as follows:

$x \circ A = b$. Where

$x = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12} \quad x_{13} \quad x_{14} \quad x_{15}]$,

$$A = \begin{bmatrix}
0.560 & 0.100 & 0.720 & 0.100 & 0.100 & 0.960 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.960 & 0.100 & 0.100 \\
0.210 & 0.100 & 0.100 & 0.300 & 0.320 & 0.100 & 0.400 & 0.100 & 0.100 & 0.100 & 0.100 & 0.630 & 0.100 & 0.700 & 0.100 & 0.800 & 0.900 & 0.360 & 0.480 & 0.560 \\
0.420 & 0.480 & 0.540 & 0.100 & 0.100 & 0.720 & 0.800 & 0.100 & 0.960 & 1.000 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 \\
0.400 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.800 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 \\
0.100 & 0.300 & 0.100 & 0.375 & 0.400 & 0.450 & 0.500 & 0.525 & 0.100 & 0.625 & 0.700 & 0.100 & 0.800 & 0.100 & 0.900 & 1.000 & 0.100 & 0.450 & 0.600 & 0.700 \\
0.350 & 0.400 & 0.450 & 0.500 & 0.100 & 0.600 & 0.100 & 0.100 & 0.800 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.600 & 0.100 & 0.100 \\
0.336 & 0.384 & 0.100 & 0.100 & 0.100 & 0.576 & 0.640 & 0.672 & 0.100 & 0.800 & 0.896 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.576 & 0.768 & 0.100 \\
0.100 & 0.375 & 0.100 & 0.100 & 0.500 & 0.100 & 0.625 & 0.100 & 0.750 & 0.100 & 0.100 & 0.100 & 1.000 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.750 & 0.875 \\
0.300 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.600 & 0.100 & 0.100 & 0.800 & 0.900 & 0.100 & 1.000 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.800 \\
0.100 & 0.100 & 0.375 & 0.100 & 0.100 & 0.500 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.875 & 0.100 & 0.100 & 1.000 & 0.100 & 0.100 & 0.500 & 0.100 & 0.100 \\
0.280 & 0.320 & 0.360 & 0.400 & 0.100 & 0.480 & 0.100 & 0.560 & 0.640 & 0.100 & 0.100 & 0.840 & 0.100 & 0.100 & 0.960 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 \\
0.100 & 0.300 & 0.100 & 0.375 & 0.400 & 0.450 & 0.500 & 0.100 & 0.100 & 0.625 & 0.100 & 0.100 & 0.800 & 0.875 & 0.100 & 1.000 & 0.100 & 0.450 & 0.600 & 0.700 \\
0.240 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.480 & 0.100 & 0.100 & 0.640 & 0.100 & 0.100 & 0.800 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 \\
0.100 & 0.100 & 0.300 & 0.100 & 0.100 & 0.400 & 0.100 & 0.100 & 0.100 & 0.100 & 0.100 & 0.700 & 0.100 & 0.100 & 0.800 & 0.100 & 1.000 & 0.100 & 0.100 & 0.100 \\
0.100 & 0.100 & 0.270 & 0.300 & 0.320 & 0.360 & 0.400 & 0.100 & 0.100 & 0.500 & 0.100 & 0.100 & 0.640 & 0.700 & 0.100 & 0.800 & 0.900 & 0.360 & 0.480 & 0.100
\end{bmatrix},$$

$$b = [0.210 \quad 0.240 \quad 0.270 \quad 0.300 \quad 0.320 \quad 0.360 \quad 0.400 \quad 0.420 \quad 0.480 \quad 0.500 \quad 0.560 \quad 0.630 \quad 0.640 \quad 0.700 \quad 0.720 \quad 0.800 \quad 0.900 \quad 0.360 \quad 0.480 \quad 0.560]$$

4. CONCLUSIONS

In this paper, we added new theoretical results for the fuzzy relational equations with max-product composition problem in finding the complete set of minimal solutions. We proposed the necessary condition possessed by a minimal solution in terms of the maximum solution, which can be easily computed. Precisely, for any minimal solution, each of its components is unique and either 0 or the corresponding component's value of the maximum solution. With the help of this necessary condition, a simple matrix includes all of minimal solutions was derived. By this simple matrix, we then proposed four rules to fix as many as possible the values of components of any minimal solution to reduce the problem. Thanks to these rules, we developed an efficient procedure to obtain the complete set of minimal solutions. The proposed procedure can obtain less the possible redundant solutions than Leotamonphong and Fang's method.

Although the necessary condition enables us to propose some rules to reduce the problem size and avoid combinatorial enumeration, it is not a universal one for general fuzzy relational equations. Actually, a similar necessary condition for fuzzy relational equations with max-min composition becomes much complicated than the one mentioned above. The following simple problem of fuzzy relational equations with max-min composition illustrates this point.

$$[x_1 \quad x_2] \circ \begin{bmatrix} 0.4 & 0.5 & 0.8 \\ 0.2 & 0.6 & 0.8 \end{bmatrix} = [0.4 \quad 0.6 \quad 0.8].$$

Direct computing shows that the maximum solution $(\bar{x}_1, \bar{x}_2) = (1, 1)$. Furthermore, the minimal solutions of this problem are $(x_1, x_2) = (0.4, 0.8)$ and $(x_1, x_2) = (0.8, 0.6)$. Note that $x_1 \neq \bar{x}_1$, $x_2 \neq \bar{x}_2$. This problem implies that such a necessary condition possessed by a minimal solution in terms of the maximum solution is not a universal one for the class of fuzzy relational equations with max-t-norm. Therefore, in the future works, we shall consider a smaller subclass of fuzzy relational equations with max-strict-t-norm to find the complete set of minimal solutions.

ACKNOWLEDGMENT

The authors would like to thank the partial support from NSC under the grant of NSC 92-2416-H-238-003 and NSC 92-2416-H-155-021, Taiwan, The Republic of China.

REFERENCES

1. Bourke, M.M. and Fisher, D.G. (1998). Solution algorithms for fuzzy relational equations with max-product composition. *Fuzzy Sets and Systems*, 94: 61-69.
2. Czogala, E., Drewniak, J., and Pedrycz, W. (1982). Fuzzy relation equations on a finite set. *Fuzzy Sets and Systems*, 7: 89-101.
3. Di Nola, A., Sessa, S., Pedrycz, W., and Sanchez, E. (1989). *Fuzzy Relational Equations and Their Applications in Knowledge Engineering*. Dordrecht: Kluwer Academic Press.
4. Fang, S.-C. and Li, G. (1999). Solving fuzzy relation equations with a linear objective function. *Fuzzy Sets and Systems*, 103: 107-113.
5. Guu, S.-M. and Wu, Y.-K. (2002). Minimizing a linear objective function with fuzzy relation equation constraints. *Fuzzy Optimization and Decision Making*, 1(4): 347-360.
6. Higashi, M. and Klir, G.J. (1984). Resolution of finite fuzzy relation equations. *Fuzzy Sets and Systems*, 13: 65- 82.
7. Lee, H.-C. and Guu, S.-M. (2003). On the optimal three-tier multimedia streaming services. *Fuzzy Optimization and Decision Making*, 2: 31-39.
8. Loetamonphong, J. and Fang, S.-C. (1999). An efficient solution procedure for fuzzy relational equations with max-product composition. *IEEE Transactions on Fuzzy Systems*, 7(4): 441-445.
9. Loetamonphong, J. and Fang, S.-C. (2001). Optimization of fuzzy relational equations with max-product composition. *Fuzzy Sets and Systems*, 118: 509-517.
10. Leotamonphong, J., Fang, S.-C., and Young., R.E. (2002). Multi-objective optimization problems with fuzzy relation equation constraints. *Fuzzy Sets and Systems*, 127: 141-164.
11. Lu, J. and Fang, S.-C. (2001). Solving nonlinear optimization problems with fuzzy relation equations constraints. *Fuzzy Sets and Systems*, 119: 1-20.
12. Pedrycz, W. (1985). On generalized fuzzy relational equations and their applications. *Journal of Mathematical Analysis and Applications*, 107: 520-536.
13. Sanchez, E. (1976). Resolution of composite fuzzy relation equations. *Information and Control*, 30: 38-48.
14. Sanchez, E. (1977). Solutions in composite fuzzy relation equations. *Fuzzy Automata and Decision Processes* (Ed.: M.M. Gupta, G.N. Saridis and B.R. Gaines). North Holland, New York, pp. 221-234.
15. Wang, H.-F. (1995). A multi-objective mathematical programming problem with fuzzy relation constraints. *Journal of Multi-Criteria Decision Analysis*, 4: 23-35.
16. Wang, P.Z., Zhang, D.Z., Sanchez, E., and Lee, E.S. (1991). Latticized linear programming and fuzzy relation inequalities. *Journal of Mathematical Analysis and Applications*, 159(1): 72-87.
17. Wu, Y.-K., Guu, S.-M., and Liu, J.Y.-C. (2002). An accelerated approach for solving fuzzy relation equations with a linear objective function. *IEEE Transactions on Fuzzy Systems*, 10(4): 552-558.