

УДК 330.4

DOI:10.24412/2782-4845-2023-7-100-123

ПОСТАНОВКА ЗАДАЧИ РЕШЕНИЯ НЕЧЕТКИХ РЕЛЯЦИОННЫХ УРАВНЕНИЙ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РЕШЕНИЙ УРАВНЕНИЙ ПРОСТЕЙШЕГО ТИПА

И.В. Черпаков, Липецкий филиал ФГОБУ ВО «Финансовый университет при Правительстве РФ», Липецк, Россия

Аннотация. Целью данного исследования является рассмотрение постановки задачи решения нечетких реляционных уравнений (НРУ), понимаемой как решение обратной задачи для нечетких соответствий, а также методы решения простейших НРУ. В качестве предмета исследования выбраны программные методы решения простейших НРУ с различными типами композиций нечетких соответствий. В работе рассмотрены расширения стандартных логических операций на единичном отрезке, различные типы композиций нечетких соответствий, использующие эти операции, приведена постановка задачи решения НРУ различных типов, приведены необходимые и достаточные условия разрешимости простейших НРУ, рассмотрены методы решения простейших НРУ через остаточные операторы в общей и аналитической форме. Приведен программный код на языке программирования Python, реализующий решение различных типов простейших НРУ.

Ключевые слова: обратная задача для нечетких соответствий, композиции нечетких соответствий, нечеткие реляционные уравнения, инвертор, t -норма, t -конорма, импликатор, остаточные операторы

Для цитирования: Черпаков И.В., Постановка задачи решения нечетких реляционных уравнений и программная реализация решений уравнений простейшего типа // ЭФО. Экономика. Финансы. Общество. 2023. №3(7) С.100-123. DOI:10.24412/2782-4845-2023-7-100-123

FORMULATION OF THE PROBLEM OF SOLVING FUZZY RELATIONAL EQUATIONS AND SOFTWARE IMPLEMENTATION OF SOLUTIONS TO EQUATIONS OF THE SIMPLEST TYPE

I.V. Cherpakov, Lipetsk Branch of FSOBU HE "Financial University under the Government of the Russian Federation", Lipetsk, Russia

Annotation. The purpose of this study is to consider the formulation of the problem of solving fuzzy relational equations (FRE), considered as a solution of the inverse problem for fuzzy correspondences, as well as methods for solving the simplest FRE. As a subject of the research the program methods of solution of the simplest FRE with different types of compositions of fuzzy correspondences are chosen. The paper considers the extensions of standard logical operations on the unit segment, different types of fuzzy correspondence compositions using these operations, the statement of the problem of solving the FREs of different types is given, the necessary and sufficient conditions of solvability of the simplest FREs are given, the methods of solving the simplest FREs through residual operators in general and analytical

form are considered. The program code in the Python programming language, which realizes the solution of various types of simplest FREs, is given.

Key words: *inverse problem for fuzzy correspondences, fuzzy correspondence compositions, fuzzy relational equations, inverter, t-norm, t-conorm, implicator, residual operators*

Введение

Обратная задача для нечетких соответствий в экономических исследованиях является одним из эффективных инструментов для анализа данных и выявления взаимосвязей между различными факторами. Она позволяет определить, какие переменные оказывают наибольшее влияние на результат, а также как они взаимодействуют друг с другом. Нечеткие соответствия могут быть использованы для описания сложных зависимостей между переменными. Например, одна переменная может влиять на другую, но при этом она также зависит от других факторов. Таким образом, нечеткие соответствия позволяют учесть все возможные зависимости между переменными и получить более точные результаты.

Одним из основных преимуществ использования нечетких соответствий является возможность учета неопределенности и неоднозначности данных. В экономических исследованиях часто встречаются ситуации, когда данные могут быть неточными или неполными, что может привести к ошибкам в анализе. Нечеткие соответствия помогают преодолеть эту проблему, позволяя учитывать возможные варианты и неопределенности. Кроме того, нечеткие соответствия могут использоваться для моделирования сложных систем. Например, они могут быть применены для описания взаимодействия между различными элементами экономической системы, такими как рынки, отрасли, предприятия и т.д. Это позволяет получить более полное представление о том, как различные факторы влияют друг на друга и какие изменения могут произойти в будущем.

Основная часть

1. Расширение стандартных логических операций

Пусть даны два четких множества X и Y , где

$$\begin{cases} X = \{x_i\}, i = \overline{1, n}; \\ Y = \{y_j\}, j = \overline{1, m}. \end{cases}$$

Нечетким соответствием R , определенным на множествах X и Y , называется нечеткое множество на декартовом произведении $X \times Y$, т. е. множество пар вида [5]

$$R = \{\mu_R(x, y), (x, y)\}, x \in X, y \in Y, \mu_R(x, y) \in [0; 1]. \quad (1)$$

Таким образом, каждой паре (x, y) из декартова произведения $X \times Y$ ставится в соответствие некоторое число из $L = [0; 1]$, определяемое как значение функции принадлежности $\mu_R(x, y)$.

Пусть даны два нечетких соответствия Q и R , определенные на декартовых произведениях множеств

$$\begin{cases} X = \{x_i\}, i = \overline{1, n}; \\ Y = \{y_j\}, j = \overline{1, m}; \\ Z = \{z_t\}, t = \overline{1, k} \end{cases} \quad (2)$$

следующим образом:

$$Q = \{\mu_Q(x_i, y_j), (x_i, y_j)\}, x_i \in X, y_j \in Y, \mu_Q(x_i, y_j) \in [0; 1], \quad (3)$$

$$R = \{\mu_R(y_j, z_t), (y_j, z_t)\}, y_j \in Y, z_t \in Z, \mu_R(y_j, z_t) \in [0; 1]. \quad (4)$$

Пусть задано нечеткое соответствие

$$S = \{\mu_S(x_i, z_t), (x_i, z_t)\}, x_i \in X, z_t \in Z, \mu_S(x_i, z_t) \in [0; 1]. \quad (5)$$

Значения $\mu_Q(x_i, y_j)$, $\mu_R(y_j, z_t)$ и $\mu_S(x_i, z_t)$ являются значениями специальных операций, вводимых на отрезке $L = [0; 1]$ — t -нормы и импликатора. Эти операции являются расширением стандартных логических операций конъюнкции и дизъюнкции соответственно. Рассмотрим кратко определения расширения стандартных логических операций. Все значения аргументов и значения операций находятся в L .

1. *Инвертор* $N(x)$ — расширение стандартной логической операции отрицания. Является унарной операцией. Для инвертора должны выполняться следующие свойства с:

$$N(0) = 1, N(1) = 0; \quad (6)$$

$$N(x_1) \leq N(x_2) \text{ при } x_1 \geq x_2; \quad (7)$$

$$N(N(x)) = x. \quad (8)$$

В качестве примера операции, которая удовлетворяет свойствам (6)–(8) может выступать так называемый *стандартный инвертор* $N(x) = 1 - x$.

2. t -норма (*треугольная норма*) $T(x, y)$ — расширение стандартной логической операции конъюнкции. Для t -нормы должны выполняться следующие свойства [2, 6]:

$$T(x, y) = T(y, x); \quad (9)$$

$$T(1, y) = y, T(0, y) = 0; \quad (10)$$

$$T(x, T(y, z)) = T(T(x, y), z); \quad (11)$$

$$T(x, y) \leq T(x, z) \text{ при } x \leq z. \quad (12)$$

Примерами t -норм могут служить логическое произведение $T(x, y) = \min(x, y)$, алгебраическое произведение $T(x, y) = xy$, t -норма Лукасевича $T(x, y) = \max(x + y - 1, 0)$. Все указанные операции удовлетворяют свойствам (9)–(12).

3. s -норма (t -конорма), — расширение стандартной логической операции дизъюнкции. Для s -нормы $S(x, y)$ должны выполняться следующие свойства [2, 6]:

$$S(x, y) = S(y, x); \quad (13)$$

$$S(1, y) = 1 \text{ и } S(0, y) = y; \quad (14)$$

$$S(x, S(y, z)) = S(S(x, y), z); \quad (15)$$

$$S(x, y) \leq S(x, z) \text{ при } x \leq z. \quad (16)$$

Примерами s -норм могут служить логическая сумма $S(x, y) = \max(x, y)$, сумма $S(x, y) = x + y - xy$, граничная сумма $S(x, y) = \min(x + y, 1)$. Указанные s -нормы удовлетворяют свойствам (13)–(16).

4. Импликатор $I(x, y)$ — расширение стандартной логической операции импликации. Для импликатора должны выполняться следующие свойства [2, 6]:

$$I(x, 1) = 1, I(1, y) = y, I(0, y) = 1; \quad (17)$$

$$I(x, y) \leq I(x, z) \text{ при } y \geq z; \quad (18)$$

$$I(x, y) \leq I(z, y) \text{ при } x \leq z. \quad (19)$$

Примерами импликаторов могут служить $I(x, y) = 1 - x + xy$, импликатор Клина-Дайнеса $I(x, y) = \max(1 - x, y)$, импликатор Лукасевича $I(x, y) = \min(1 - x + y, 1)$. Указанные импликаторы удовлетворяют свойствам (17)–(19).

Можно показать [например, 9], что функции

$$N(T(x, N(y))) \quad (20)$$

и

$$S(N(x), y) \quad (21)$$

удовлетворяют свойствам (17)–(19) для любых $N(x)$, $T(x, y)$ и $S(x, y)$.

Таким образом, для заданного инвертора $N(x)$ и t -нормы $T(x, y)$ функция, соответствующая выражению (20), а именно,

$$I_{N,T}(x, y) = N(T(x, N(y))), \quad (22)$$

представляет собой импликатор (говорят, что инвертор и t -норма индуцируют импликатор).

Для заданного инвертора $N(x)$ и s -нормы $S(x, y)$ функция, соответствующая выражению (21), а именно,

$$I_{N,S}(x, y) = S(N(x), y), \quad (23)$$

представляет собой импликатор (говорят, что инвертор и s -норма индуцируют импликатор).

Например, импликатор Клина-Дайнеса индуцирован стандартным инвертором и логической суммой, а импликатор Лукасевича индуцирован стандартным инвертором и t -нормой Лукасевича.

Выражения (20)–(21) позволяют конструировать неограниченное число импликаторов при заданных инверторе, t -норме и s -норме.

2. Композиции нечетких соответствий

Рассмотрим базовые понятия, связанные с определением композиций нечетких соответствий [6, 11].

Композицией (круговой композицией) двух нечетких соответствий Q и R называется нечеткое соответствие, обозначаемое как $Q \circ R$ и определяемое следующим образом:

$$Q \circ R(x_i, z_t) = \max_{y_j \in Y} \left\{ T\left(\mu_Q(x_i, y_j), \mu_R(y_j, z_t)\right) \right\}, \quad (24)$$

где T — t -норма.

Субкомпозицией двух нечетких соответствий Q и R называется нечеткое соответствие, обозначаемое как $Q \triangleleft R$ и определяемое следующим образом:

$$Q \triangleleft R(x_i, z_t) = \min_{y_j \in Y} \left\{ I\left(\mu_Q(x_i, y_j), \mu_R(y_j, z_t)\right) \right\}, \quad (25)$$

где I — импликатор.

Суперкомпозицией двух нечетких соответствий Q и R называется нечеткое соответствие, обозначаемое как $Q \triangleleft R$ и определяемое следующим образом:

$$Q \triangleright R(x_i, z_t) = \min_{y_j \in Y} \left\{ I\left(\mu_Q(y_j, z_t), \mu_R(x_i, y_j)\right) \right\}, \quad (26)$$

где I — импликатор.

В случае, если композиция нечетких соответствий определяется по формуле (24), говорят, что имеет место \max - T композиция. В случае, если композиция нечетких соответствий определяется по формулам (25) или (26), говорят, что имеет место \min - I композиция.

3. Постановка обратной задачи для композиции нечетких соответствий в форме нечеткого реляционного уравнения

Пусть дана композиция нечетких соответствий

$$Q \otimes R = S, \quad (27)$$

где \otimes — композиция вида $\max-T$ или $\min-I$.

Обратной задачей для композиции нечетких соответствий вида (27) называется нахождение одного из следующих нечетких соответствий:

- Q при известных R, S и \otimes ;
- R при известных Q, S и \otimes .

Запись обратной задачи в математической форме представляет собой нечеткое реляционное уравнение (НРУ)— матричное уравнение с неизвестным X вида

$$X \otimes R = S \quad (28)$$

или

$$Q \otimes X = S. \quad (29)$$

Композиция вида $\otimes = \max-T$ обладает свойством коммутативности, поэтому уравнение вида (28) эквивалентно уравнению

$$R \otimes X = S, \quad (30)$$

т. е. для $\otimes = \max-T$ уравнения (28) и (29) эквивалентны.

Композиция вида $\otimes = \min-I$ не обладает свойством коммутативности, поэтому для данного типа композиции уравнения (28) и (29) различны. При этом уравнение вида (28) называют *левым $\min-I$ уравнением*, уравнение (29) — *правым $\min-I$ уравнением*.

В зависимости от размерности матриц, составляющих уравнения (28) и (29), выделяют различные типы НРУ [9]:

1. Уравнение общего вида, в которых Q, R и S являются матрицами.

Левое уравнение общего вида (соответствующее (28)) выглядит следующим образом:

$$\begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nm} \end{pmatrix} \otimes \begin{pmatrix} \mu_R(y_1, z_1) & \cdots & \mu_R(y_1, z_t) & \cdots & \mu_R(y_1, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_R(y_j, z_1) & \cdots & \mu_R(y_j, z_t) & \cdots & \mu_R(y_j, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_R(y_m, z_1) & \cdots & \mu_R(y_m, z_t) & \cdots & \mu_R(y_m, z_k) \end{pmatrix} =$$

$$= \begin{pmatrix} \mu_S(x_1, z_1) & \cdots & \mu_S(x_1, z_t) & \cdots & \mu_S(x_1, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_S(x_i, z_1) & \cdots & \mu_S(x_i, z_t) & \cdots & \mu_S(x_i, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_S(x_n, z_1) & \cdots & \mu_S(x_n, z_t) & \cdots & \mu_S(x_n, z_k) \end{pmatrix} \quad (31)$$

с неизвестным

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nm} \end{pmatrix}.$$

Правое НРУ общего вида (соответствующее (29)) выглядит следующим образом:

$$\begin{pmatrix} \mu_Q(x_1, y_1) & \cdots & \mu_Q(x_1, y_j) & \cdots & \mu_Q(x_1, y_m) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_Q(x_i, y_1) & \cdots & \mu_Q(x_i, y_j) & \cdots & \mu_Q(x_i, y_m) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_Q(x_n, y_1) & \cdots & \mu_Q(x_n, y_j) & \cdots & \mu_Q(x_n, y_m) \end{pmatrix} \otimes \begin{pmatrix} x_{11} & \cdots & x_{1t} & \cdots & x_{1k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{j1} & \cdots & x_{jt} & \cdots & x_{jk} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{m1} & \cdots & x_{mt} & \cdots & x_{mk} \end{pmatrix} =$$

$$= \begin{pmatrix} \mu_S(x_1, z_1) & \cdots & \mu_S(x_1, z_t) & \cdots & \mu_S(x_1, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_S(x_i, z_1) & \cdots & \mu_S(x_i, z_t) & \cdots & \mu_S(x_i, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_S(x_n, z_1) & \cdots & \mu_S(x_n, z_t) & \cdots & \mu_S(x_n, z_k) \end{pmatrix}$$

(32)

с неизвестным

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1t} & \cdots & x_{1k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{j1} & \cdots & x_{jt} & \cdots & x_{jk} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{m1} & \cdots & x_{mt} & \cdots & x_{mk} \end{pmatrix}.$$

2. *Множество систем линейных уравнений.* В этом случае неизвестное нечеткое соответствие из левой части уравнения представляет собой матрицу, а известное нечеткое соответствие — вектор-столбец (для НРУ вида (28)) или вектор-строку (для НРУ вида (29)). Множество систем левых линейных НРУ имеет вид

$$\begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nm} \end{pmatrix} \otimes \begin{pmatrix} \mu_R(y_1, z_1) \\ \cdots \\ \mu_R(y_j, z_1) \\ \cdots \\ \mu_R(y_m, z_1) \end{pmatrix} = \begin{pmatrix} \mu_S(x_1, z_1) \\ \cdots \\ \mu_S(x_i, z_1) \\ \cdots \\ \mu_S(x_n, z_1) \end{pmatrix} \quad (33)$$

с неизвестным

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1m} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{nm} \end{pmatrix}.$$

Множество систем правых НРУ имеет вид

$$\begin{pmatrix} \mu_Q(x_1, y_1) & \cdots & \mu_Q(x_1, y_j) & \cdots & \mu_Q(x_1, y_m) \end{pmatrix} \otimes \begin{pmatrix} x_{11} & \cdots & x_{1t} & \cdots & x_{1k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{j1} & \cdots & x_{jt} & \cdots & x_{jk} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{m1} & \cdots & x_{mt} & \cdots & x_{mk} \end{pmatrix} = \begin{pmatrix} \mu_S(x_1, z_1) & \cdots & \mu_S(x_1, z_t) & \cdots & \mu_S(x_1, z_k) \end{pmatrix} \quad (34)$$

с неизвестным

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1t} & \cdots & x_{1k} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{j1} & \cdots & x_{jt} & \cdots & x_{jk} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{m1} & \cdots & x_{mt} & \cdots & x_{mk} \end{pmatrix}.$$

НРУ вида (33) и (34) представляют собой частные случаи НРУ вида (31) и (32) соответственно при $k = 1$ (НРУ вида (33)) и при $n = 1$ (НРУ вида (34)), т. е. когда четкие множества Z и X при задании (2) представляют собой одноэлементные множества: $Z = \{z_1\}$, $X = \{x_1\}$.

3. Система линейных уравнений. В этом случае неизвестное нечеткое соответствие представляет собой вектор-строку (для левых НРУ) или вектор-столбец (для правых НРУ). При этом второе нечеткое соответствие, с которым рассматривается композиция неизвестного, представляет собой матрицу.

Система левых линейных НРУ имеет вид

$$\begin{aligned} & \left(x_{11} \quad \cdots \quad x_{1j} \quad \cdots \quad x_{1m} \right) \otimes \begin{pmatrix} \mu_R(y_1, z_1) & \cdots & \mu_R(y_1, z_t) & \cdots & \mu_R(y_1, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_R(y_j, z_1) & \cdots & \mu_R(y_j, z_t) & \cdots & \mu_R(y_j, z_k) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_R(y_m, z_1) & \cdots & \mu_R(y_m, z_t) & \cdots & \mu_R(y_m, z_k) \end{pmatrix} = \\ & = \left(\mu_S(x_1, z_1) \quad \cdots \quad \mu_S(x_1, z_t) \quad \cdots \quad \mu_S(x_1, z_k) \right) \end{aligned} \quad (35)$$

с неизвестным

$$X = \left(x_{11} \quad \cdots \quad x_{1j} \quad \cdots \quad x_{1m} \right).$$

Система правых линейных НРУ имеет вид

$$\begin{pmatrix} \mu_Q(x_1, y_1) & \cdots & \mu_Q(x_1, y_j) & \cdots & \mu_Q(x_1, y_m) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_Q(x_i, y_1) & \cdots & \mu_Q(x_i, y_j) & \cdots & \mu_Q(x_i, y_m) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_Q(x_n, y_1) & \cdots & \mu_Q(x_n, y_j) & \cdots & \mu_Q(x_n, y_m) \end{pmatrix} \otimes \begin{pmatrix} x_{11} \\ \cdots \\ x_{i1} \\ \cdots \\ x_{m1} \end{pmatrix} = \begin{pmatrix} \mu_S(x_1, z_1) \\ \cdots \\ \mu_S(x_i, z_1) \\ \cdots \\ \mu_S(x_n, z_1) \end{pmatrix} \quad (36)$$

с неизвестным

$$X = \left(x_{11} \quad \cdots \quad x_{i1} \quad \cdots \quad x_{m1} \right)^T,$$

где T — символ транспонирования.

НРУ вида (35) и (36) представляют собой частные случаи НРУ вида (31) и (32) соответственно при $n = 1$ (НРУ вида (35)) и $k = 1$ (НРУ вида (36)), т. е. когда четкие множества X и Z при задании (2) представляют собой одноэлементные множества: $X = \{x_1\}$ для левого НРУ, $Z = \{z_1\}$ для правого НРУ.

4. *Линейное (полиномиальное) НРУ.* В этом случае неизвестное нечеткое соответствие X представляет собой вектор-строку (для левого линейного НРУ) или вектор-столбец (для правого линейного НРУ). При этом второе нечеткое соответствие, в композиции с которым рассматривают неизвестное, представляется как вектор-столбец (для левого линейного НРУ) или как вектор-строка (для правого НРУ). Правая часть уравнения представляет собой скалярное значение — результат композиции нечетких соответствий в виде вектора-строки и вектора-столбца.

Левое линейное НРУ имеет вид

$$\begin{aligned} & \left(x_{11} \quad \cdots \quad x_{1j} \quad \cdots \quad x_{1m} \right) \otimes \left(\mu_R(y_{11}, z_1) \quad \cdots \quad \mu_R(y_{1j}, z_1) \quad \cdots \quad \mu_R(y_{1m}, z_1) \right)^T = \\ & = \mu_S(x_1, z_1) \end{aligned} \quad (37)$$

с неизвестным

$$X = \left(x_{11} \quad \cdots \quad x_{1j} \quad \cdots \quad x_{1m} \right).$$

Правое линейное НРУ имеет вид

$$\begin{aligned} (\mu_Q(x_1, y_1) \cdots \mu_Q(x_1, y_j) \cdots \mu_Q(x_1, y_m)) \otimes (x_{11} \cdots x_{j1} \cdots x_{m1})^T = \\ = \mu_S(x_1, z_1) \end{aligned} \quad (38)$$

с неизвестным

$$X = (x_{11} \cdots x_{j1} \cdots x_{m1})^T.$$

НРУ вида (37) и (38) представляют собой частные случаи НРУ вида (31) и (32) соответственно при $n = k = 1$, т. е. когда четкие множества X и Z при задании (2) одновременно представляют собой одноэлементные множества: $X = \{x_1\}$, $Z = \{z_1\}$.

5. *Простейшее (атомарное, элементарное) НРУ.* Представляют собой наиболее простой случай, когда все множества из (2) являются одноэлементными, т. е. когда $n = m = k = 1$.

Далее рассмотрим подробное решение простейших НРУ и приведем программную реализацию решений.

4. Необходимые и достаточные условия разрешимости простейших НРУ

В простейших уравнениях все матрицы нечетких соответствий имеют размерность 1×1 , т. е. представляют собой скалярные значения. В формальной записи НРУ с различными типами композиций примут вид

$$T(x, r) = s, \quad (39)$$

$$I(x, r) = s, \quad (40)$$

$$I(q, x) = s. \quad (41)$$

В зависимости от значений r и s НРУ вида (39)–(41) могут быть не разрешимыми. Необходимые и достаточные условия, при которых НРУ разрешимы, приведены в табл. 1.

Таблица 1. Решение простейших НРУ для различных t -норм*

Уравнение	Необходимые и достаточные условия разрешимости
$T(x, r) = s$	$r \geq s$
$I(x, r) = s$	$r \leq s$
$I(q, x) = s$	$N(q) \leq s$, где $N(x)$ — инвертор, через который индуцирован импликатор

* составлено автором на основе [9]

5. Решение простейших НРУ с \max - T композицией с использованием остаточных операторов

Решение НРУ вида (39) может быть найдено с использованием *остаточных операторов* (residual operators) [9, 10] вида

$$T^-(r, s) = \min \{z \in [0, 1] : T(z, r) \geq s\}, \quad (42)$$

$$T^+(r, s) = \max \{z \in [0, 1] : T(z, r) \leq s\}. \quad (43)$$

Решением НРУ вида (39) будет отрезок

$$\left[T^-(r, s); T^+(r, s) \right]. \quad (44)$$

Рассмотрим решение НРУ вида (39) для трех наиболее часто применяемых на практике t -норм, а именно, для операции минимума $T(x, y) = \min(x, y)$, алгебраического произведения $T(x, y) = xy$ и t -нормы Лукасевича $T(x, y) = \max(x + y - 1, 0)$.

Пример 1. Реализуем остаточные операторы в форме (42) и (43) на языке программирования Python. Единичный отрезок разбивается на 10^5 точек (включая концы отрезка) и определяются $T^-(r, s)$ и $T^+(r, s)$, обозначенные в программном коде как `t_minus` и `t_plus` соответственно. Для удобства восприятия результата получаемые значения $T^-(r, s)$ и $T^+(r, s)$ округлены до четырех знаков после десятичной точки при помощи функции `round()`. В случае отсутствия решения программа выводит значение **None**.

Программный код реализации остаточных операторов в форме (42) и (43) приведен в листинге 1. Реализация выполнена с использованием технологий программирования на языках высокого уровня [7, 8].

Листинг 1. Реализация остаточных операторов для уравнений $T(x, r) = s$ *

```
import numpy as np

r = 0.7
s = 0.2
t = np.linspace(0, 1, 10**5)

# T(x, y) = min(x, y)
t_minus = [z for z in t if min(z, r) >= s]
t_minus = round(min(t_minus), 4) if t_minus else None
t_plus = [z for z in t if min(z, r) <= s]
t_plus = round(max(t_plus), 4) if t_plus else None
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)
```

```

# T(x, y) = xy
t_minus = [z for z in t if z * r >= s]
t_minus = round(min(t_minus), 4) if t_minus else None
t_plus = [z for z in t if z * r <= s]
t_plus = round(max(t_plus), 4) if t_plus else None
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)

# T(x, y) = max(x + y - 1, 0)
t_minus = [z for z in t if max(z + r - 1, 0) >= s]
t_minus = round(min(t_minus), 4) if t_minus else None
t_plus = [z for z in t if max(z + r - 1, 0) <= s]
t_plus = round(max(t_plus), 4) if t_plus else None
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

Рассмотрим решение различных простейших НРУ при следующих значениях r и s :

— $r = 0.8$, $s = 0.1$. Имеем уравнение $T(x, 0.8) = 0.1$. Необходимое и достаточное условие разрешимости $r \geq s$ выполняется, следовательно уравнение разрешимо;

— $r = 0.6$, $s = 0.6$. Имеем уравнение $T(x, 0.6) = 0.6$. Необходимое и достаточное условие разрешимости $r \geq s$ выполняется, следовательно уравнение разрешимо;

— $r = 0.4$, $s = 0.6$. Имеем уравнение $T(x, 0.4) = 0.6$. Необходимое и достаточное условие разрешимости $r \geq s$ не выполняется, следовательно уравнение не разрешимо.

Для указанных значений r и s полученные в результате работы программы листинга 1 приведены в табл. 2.

Таблица 2. Решение простейших НРУ для различных t -норм*

t -норма	r	s	Решение
$\min(x, y)$	0.8	0.1	[0.1; 0.1]
	0.6	0.6	[0.6; 1.0]
	0.4	0.6	None

Продолжение Таблицы 2.

t -норма	r	s	Решение
xy	0.8	0.1	[0.125;0.125]
	0.6	0.6	[1.0;1.0]
	0.4	0.6	None
$\max(x + y - 1, 0)$	0.8	0.1	[0.3;0.3]
	0.6	0.6	[1.0;1.0]
	0.4	0.6	None

* составлено автором на основе собственных расчетов

Реализованные остаточные операторы могут быть использованы в дальнейшем для решения более сложных типов НРУ.

6. Решение простейших НРУ с \max - T композицией с использованием остаточных операторов в аналитической форме

Использование остаточных операторов в форме (42) и (43) возможно, хотя и требует вычислительных ресурсов в случае большого количества значений. Поэтому на практике, когда имеют дело с t -нормами $T(x, y) = \min(x, y)$, $T(x, y) = xy$ и $T(x, y) = \max(x + y - 1, 0)$, используют более простые формы их определения. Они получены как результат аналитического решения уравнений вида $\min(x, r) = s$, $x \cdot r = s$ и $\max(x + r - 1, 0) = s$ соответственно. Остаточные операторы (42) и (43) в аналитической форме для рассматриваемых t -норм приведены в табл. 3 [9, 10].

Таблица 3. Аналитическая форма остаточных операторов для t -норм*

t -норма	$T^-(r, s)$	$T^+(r, s)$
$\min(x, y)$	$\begin{cases} \emptyset, & \text{если } r < s; \\ s, & \text{иначе.} \end{cases}$	$\begin{cases} 1, & \text{если } r \leq s; \\ s, & \text{иначе.} \end{cases}$
xy	$\begin{cases} \emptyset, & \text{если } r < s; \\ s/r, & \text{если } 0 < s \leq r; \\ 0, & \text{иначе.} \end{cases}$	$\begin{cases} 1, & \text{если } r \leq s; \\ s/r, & \text{иначе.} \end{cases}$
$\max(x + y - 1, 0)$	$\begin{cases} \emptyset, & \text{если } r < s; \\ 1 - r + s, & \text{если } 0 < s \leq r; \\ 0, & \text{иначе.} \end{cases}$	$\begin{cases} 1, & \text{если } r \leq s; \\ 1 - r + s, & \text{если } 0 < s \leq r; \\ 1 - r, & \text{иначе.} \end{cases}$

* составлено автором на основе [9, 10]

Если значение $T^-(r,s)$ не существует (равно пустому множеству \emptyset), то решения НРУ вида (39) не существует. В этом случае значения r и s не удовлетворяют необходимым и достаточным условиям разрешимости $\max-T$ уравнений, приведенным в табл. 1.

Пример 2. Рассмотрим решение НРУ вида (39) при $r=s=0.6$, т. е. уравнение $T(x,0.6)=0.6$. Необходимое и достаточное условие разрешимости $r \geq s$ выполняется, следовательно, уравнение разрешимо.

Реализация решений через остаточные операторы из табл. 3 приведена в листинге 2.

Листинг 2. Реализация остаточных операторов в аналитической форме для уравнений вида $T(x,r)=s^*$

```

r = 0.6
s = 0.6

# T(x, y) = min(x, y)
t_minus = None if r < s else s
t_plus = 1 if r <= s else s
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)

# T(x, y) = xy
t_minus = None if r < s else s / r if 0 < s <= r else 0
t_plus = 1 if r <= s else s / r
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)

# T(x, y) = max(x + y - 1, 0)
t_minus = None if r < s else 1 - r + s if 0 < s <= r else 0
t_plus = 1 if r <= s else 1 - r + s if 0 < s <= r else 1-r
result = [t_minus, t_plus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

В результате имеем для $T(x,y) = \min(x,y)$ решение $[0.6;1]$. Для $T(x,y) = xy$ и $T(x,y) = \max(x+y-1,0)$ решениями будут отрезки $[1;1]$, вырождающиеся в скалярное значение 1.

7. Решение простейших НРУ с $\min-I$ композицией с использованием остаточных операторов

НРУ вида (40) и (41) решаются аналогично НРУ типа $\max-T$. Остаточные операторы для них определяются следующим образом [9]:

$$I_l^-(r, s) = \min \{z \in [0, 1] : I(z, r) \leq s\}, \quad (45)$$

$$I_l^+(r, s) = \max \{z \in [0, 1] : I(z, r) \geq s\}, \quad (46)$$

$$I_r^-(q, s) = \min \{z \in [0, 1] : I(q, z) \geq s\}, \quad (47)$$

$$I_r^+(q, s) = \max \{z \in [0, 1] : I(q, z) \leq s\}. \quad (48)$$

Решением левых $\min-I$ уравнений (вида (40)) будет отрезок

$$\left[I_l^-(r, s); I_l^+(r, s) \right], \quad (49)$$

решением правых $\min-I$ уравнений (вида (41)) будет отрезок

$$\left[I_r^-(q, s); I_r^+(q, s) \right]. \quad (50)$$

Если для левого $\min-I$ уравнения хотя бы одно из значений $I_l^-(r, s)$ или $I_l^+(r, s)$ не существует (равно \emptyset), то решения НРУ вида (40) не существует. Аналогично, если для правого $\min-I$ уравнения хотя бы одно из значений $I_r^-(q, s)$ или $I_r^+(q, s)$ не существует (равно пустому множеству \emptyset), то решения НРУ вида (41) не существует.

Рассмотрим решение НРУ вида (40) для трех наиболее часто применяемых на практике импликаторов, а именно, $I(x, y) = \max(1 - x, y)$, $I(x, y) = 1 - x + x \cdot y$ и $I(x, y) = \min(1 - x + y, 1)$.

Пример 3. Реализуем остаточные операторы в формах (45)–(48) на языке программирования Python. Как и в случае НРУ вида (39), единичный отрезок разбивается на 10^5 точек (включая концы единичного отрезка). Определяются $I_l^-(r, s)$ и $I_l^+(r, s)$, обозначенные в программном коде как `i_lminus`, `i_lplus` соответственно. Для удобства восприятия результата получаемые значения округлены до четырех знаков после десятичной точки при помощи функции `round()`.

Программный код реализации остаточных операторов в форме (45) и (46) приведен в листинге 3.

Листинг 3. Реализация остаточных операторов для уравнений вида $I(x, r) = s^*$

```
import numpy as np

r = 0.3; s = 0.3
t = np.linspace(0, 1, 10**5)
```

```

# I(x, y) = max(1 - x, y)
i_lminus = [z for z in t if max(1 - z, r) <= s]
i_lminus = round(min(i_lminus), 4) if i_lminus else None
i_lplus = [z for z in t if max(1 - z, r) >= s]
i_lplus = round(max(i_lplus), 4) if i_lplus else None
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = 1 - x + x * y
i_lminus = [z for z in t if 1 - z + z * r <= s]
i_lminus = round(min(i_lminus), 4) if i_lminus else None
i_lplus = [z for z in t if 1 - z + z * r >= s]
i_lplus = round(max(i_lplus), 4) if i_lplus else None
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = min(1 - x + y, 1)
i_lminus = [z for z in t if min(1 - z + r, 1) <= s]
i_lminus = round(min(i_lminus), 4) if i_lminus else None
i_lplus = [z for z in t if min(1 - z + r, 1) >= s]
i_lplus = round(max(i_lplus), 4) if i_lplus else None
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

Рассмотрим решение различных простейших левых min- I НРУ при следующих значениях r и s :

— $r = 0.4$, $s = 0.8$. Имеем уравнение $I(x, 0.4) = 0.8$. Необходимое и достаточное условие разрешимости $r \leq s$ выполняется, следовательно уравнение разрешимо;

— $r = 0.5$, $s = 0.5$. Имеем уравнение $I(x, 0.5) = 0.5$. Необходимое и достаточное условие разрешимости $r \leq s$ выполняется, следовательно уравнение разрешимо;

— $r = 0.6$, $s = 0.2$. Имеем уравнение $I(x, 0.6) = 0.2$. Необходимое и достаточное условие разрешимости $r \leq s$ не выполняется, следовательно уравнение не разрешимо.

Для указанных значений r и s полученные в результате работы программы листинга 3 приведены в табл. 4.

Таблица 4. Решение простейших левых НРУ для различных импликаторов*

Импликатор	r	s	Решение
$I(x, y) = \max(1 - x, y)$ и	0.4	0.8	[0.2; 0.2]
	0.5	0.5	[0.5; 1.0]
	0.6	0.2	None
$I(x, y) = 1 - x + x \cdot y$	0.4	0.8	[0.33; 0.33]
	0.5	0.5	[1.0; 1.0]
	0.6	0.2	None
$I(x, y) = \min(1 - x + y, 1)$	0.4	0.8	[0.6; 0.6]
	0.5	0.5	[1.0; 1.0]
	0.6	0.2	None

* составлено автором на основе собственных расчетов

Программный код реализации остаточных операторов в форме (47) и (48) приведен в листинге 4. Определяются $I_r^-(q, s)$, $I_r^+(q, s)$, обозначенные в программном коде как `i_rminus` и `i_rplus` соответственно. Для округления значений, как и ранее, использована функция `round()`.

Листинг 4. Реализация остаточных операторов для уравнений вида $I(q, x) = s$ *

```
import numpy as np

q = 0.3; s = 0.7
t = np.linspace(0, 1, 10**5)

# I(x, y) = max(1 - x, y)
i_rminus = [z for z in t if max(1 - q, z) >= s]
i_rminus = round(min(i_rminus), 4) if i_rminus else None
i_rplus = [z for z in t if max(1 - q, z) <= s]
i_rplus = round(max(i_rplus), 4) if i_rplus else None
result = [i_rminus, i_rplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = 1 - x + x * y
i_rminus = [z for z in t if 1 - q + q * z >= s]
i_rminus = round(min(i_rminus), 4) if i_rminus else None
i_rplus = [z for z in t if 1 - q + q * z <= s]
i_rplus = round(max(i_rplus), 4) if i_rplus else None
result = [i_rminus, i_rplus]
if None in result:
```

```

    print(None)
else:
    print(result)

# I(x, y) = min(1 - x + y, 1)
i_rminus = [z for z in t if min(1 - q + z, 1) >= s]
i_rminus = round(min(i_rminus), 4) if i_rminus else None
i_rplus = [z for z in t if min(1 - q + z, 1) <= s]
i_rplus = round(max(i_rplus), 4) if i_rplus else None
result = [i_rminus, i_rplus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

Рассмотрим решение различных простейших правых \min -I НРУ при следующих значениях r и s :

— $q = 0.3$, $s = 0.7$. Имеем уравнение $I(0.3, x) = 0.7$. Все рассматриваемые в данной статье импликаторы индуцированы стандартным инвертором и различными t -нормами. Необходимое и достаточное условие разрешимости, приведенные в табл. 1, $N(q) \leq s$ или $1 - q = 1 - 0.3 = 0.7 \leq 0.7$ выполняется, следовательно уравнение разрешимо;

— $q = 0.6$, $s = 0.6$. Имеем уравнение $I(0.6, x) = 0.6$. Необходимое и достаточное условие разрешимости $N(q) \leq s$ или $1 - q = 1 - 0.6 = 0.4 \leq 0.6$ выполняется, следовательно уравнение разрешимо;

— $q = 0.6$, $s = 0.1$. Имеем уравнение $I(0.6, x) = 0.1$. Необходимое и достаточное условие разрешимости $1 - q = 1 - 0.6 = 0.4 > 0.1$ не выполняется, следовательно уравнение не разрешимо.

Для указанных значений q и s полученные в результате работы программы листинга 3 приведены в табл. 5.

Таблица 5. Решение простейших правых НРУ для различных импликаторов*

Импликатор	q	s	Решение
$\max(1 - x, y)$	0.3	0.7	[0.0; 0.7]
	0.6	0.6	[0.6; 0.6]
	0.6	0.1	None
$1 - x + x \cdot y$	0.3	0.7	[0.0; 0.0]
	0.6	0.6	[0.33; 0.33]
	0.6	0.1	None

Продолжение Таблицы 5.

Импликатор	q	s	Решение
$\min(1-x+y, 1)$	0.3	0.7	[0.0; 0.0]
	0.6	0.6	[0.2; 0.2]
	0.6	0.1	None

* составлено автором на основе собственных расчетов

Реализованные остаточные операторы могут быть использованы в дальнейшем для решения более сложных типов левых и правых $\min-I$ НРУ.

8. Решение простейших НРУ с $\min-I$ композицией с использованием остаточных операторов в аналитической форме

Для левых $\min-I$ уравнений остаточные операторы вида (45)–(46) для рассматриваемых импликаторов $I(x, y) = \max(1-x, y)$, $I(x, y) = 1-x+x \cdot y$ и $I(x, y) = \min(1-x+y, 1)$ можно задать аналитически. Это упрощает вычисления при программной реализации решений НРУ. Аналитические формы задания остаточных операторов (45)–(46) приведены в табл. 6 [9, 10].

Таблица 6. Аналитическая форма остаточных операторов для левых $\min-I$ уравнений*

Импликатор	$I_l^-(r, s)$	$I_l^+(r, s)$
$\max(1-x, y)$	$\begin{cases} \emptyset, & \text{если } r > s; \\ 1-s, & \text{иначе.} \end{cases}$	$\begin{cases} 1, & \text{если } r \geq s; \\ 1-s, & \text{иначе.} \end{cases}$
$1-x+x \cdot y$	$\begin{cases} \emptyset, & \text{если } r > s; \\ \frac{s-1}{r-1}, & \text{если } r \leq s \text{ и } r \neq 1; \\ 0, & \text{иначе.} \end{cases}$	$\begin{cases} 1, & \text{если } r \geq s; \\ \frac{s-1}{r-1}, & \text{иначе.} \end{cases}$
$\min(1-x+y, 1)$	$\begin{cases} \emptyset, & \text{если } r > s; \\ 1-s+r, & \text{если } r \leq s < 1; \\ 0, & \text{иначе.} \end{cases}$	$\min(1-s+r, 1)$

* составлено автором на основе [9, 10]

Для правых $\min-I$ уравнений остаточные операторы вида (47)–(48) для рассматриваемых импликаторов так же можно задать аналитически. Формы задания приведены в табл. 7 [9, 10].

Таблица 7. Аналитическая форма остаточных операторов для правых $\min-I$ уравнений*

Импликатор	$I_r^-(q, s)$	$I_r^+(q, s)$
$\max(1-x, y)$	$\begin{cases} 0, & \text{если } q+s \leq 1; \\ s, & \text{иначе.} \end{cases}$	$\begin{cases} \emptyset, & \text{если } q+s < 1; \\ s, & \text{иначе.} \end{cases}$
$1-x+x \cdot y$	$\begin{cases} 0, & \text{если } q+s \leq 1; \\ \frac{q+s-1}{q}, & \text{иначе.} \end{cases}$	$\begin{cases} \emptyset, & \text{если } q+s < 1; \\ \frac{q+s-1}{q}, & \text{если } q+s \geq 1 \text{ и } q \neq 0; \\ 1, & \text{иначе.} \end{cases}$
$\min(1-x+y, 1)$	$\max(q+s-1, 0)$	$\begin{cases} \emptyset, & \text{если } q+s < 1; \\ q+s-1, & \text{если } 1-q \leq s < 1; \\ 1, & \text{иначе.} \end{cases}$

* составлено автором на основе [9, 10]

Пример 4. Рассмотрим решение НРУ вида (40) при $r=0.4$, $s=0.8$, т. е. уравнение $I(x, 0.4)=0.8$. Реализация решений через остаточные операторы из табл. 6 приведена в листинге 5.

Листинг 5. Реализация остаточных операторов в аналитической форме для уравнений вида $I(x, r)=s$ *

```

r = 0.4
s = 0.8

# I(x, y) = max(1 - x, y)
i_lminus = None if r > s else 1 - s
i_lplus = 1 if r >= s else 1 - s
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = 1 - x + x * y
i_lminus = None if r > s else (s - 1) / (r - 1) \
if (r <= s) and (r != 1) else 0
i_lplus = 1 if r >= s else (s - 1) / (r - 1)
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = min(1 - x + y, 1)
i_lminus = None if r > s else 1 - s + r \
if r <= s < 1 else 0

```

```

i_lplus = min(1 - s + r, 1)
result = [i_lminus, i_lplus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

Результаты работы программы листинга 5 приведены ниже

```

[0.19999999999999996, 0.19999999999999996]
[0.33333333333333326, 0.33333333333333326]
[0.6, 0.6]

```

В первых двух строках получены результаты типа **float** в виде бесконечных десятичных дробей, что связано со спецификой реализации деления в языке Python. При округлении, например, с помощью функции **round()**, результат соответствует значениям из табл. 4 при $r = 0.4$, $s = 0.8$.

Пример 5. Рассмотрим решение правых \min -I НРУ вида (41) при $q = 0.3$, $s = 0.7$, т. е. уравнение $I(0.3, x) = 0.7$. Реализация решений через остаточные операторы из табл. 7 приведена в листинге 6.

Листинг 6. Реализация остаточных операторов в аналитической форме для уравнений вида $I(q, x) = s^*$

```

q = 0.3
s = 0.7

```

```

# I(x, y) = max(1 - x, y)
i_rminus = 0 if q + s <= 1 else s
i_rplus = None if s < 1 - q else 1 - q\
if s == 1 - q else s
result = [i_rminus, i_rplus]
if None in result:
    print(None)
else:
    print(result)

# I(x, y) = 1 - x + x * y
i_rminus = 0 if q + s <= 1 else (q + s - 1) / q
i_rplus = None if q + s < 1 else (q + s - 1) / q\
    if (q + s >= 1) and (q != 0) else 1
result = [i_rminus, i_rplus]
if None in result:
    print(None)
else:
    print(result)

```

```

# I(x, y) = min(1 - x + y, 1)
i_rminus = max(q + s - 1, 0)

```

```

i_rplus = None if q + s < 1 else (q + s - 1)\
if (1 - q) <= s < 1 else 1
result = [i_rminus, i_rplus]
if None in result:
    print(None)
else:
    print(result)

```

** разработано автором*

Результаты работы программы листинга 6 приведены ниже.

```

[0, 0.7]
[0, 0.0]
[0.0, 0.0]

```

Полученные результаты совпадают с результатами, приведенными в табл. 5.

Программная реализация нахождения остаточных операторов в аналитической форме проще и быстрее, однако это верно только для достаточно простых расширений стандартных логических операций. В случае, когда аналитически выразить остаточные операторы не получается, прибегают к численному решению (по формулам (42)–(43) и (45)–(48)).

Заключение

Рассмотренные в данной статье методы решения простейших НРУ и их программная реализация является лишь начальным этапом для решения НРУ общего вида. Исходя из изложенного теоретического материала и результатов программной реализации можно сделать ряд выводов:

1. Каждый из типов НРУ в порядке возрастания сложности (простейшие, линейные, системы линейных НРУ, множество систем линейных НРУ, уравнения общего вида) имеет решение, связанное с решением более простого типа НРУ.

2. Многообразие расширений стандартных логических операций приводит к ограниченности аналитического решения даже простейших НРУ.

3. Перед решением любого типа НРУ следует проверить выполнимость необходимых и достаточных условий разрешимости НРУ.

4. Решение $\max-T$ и левых $\min-I$ уравнений схожи. Решение правых $\min-I$ уравнений обладает двойственностью по отношению к методам для $\max-T$ и левым $\min-I$ уравнениям: операция минимума заменяется на максимум и наоборот, условие «меньше или равно» заменяется на «больше или равно» и наоборот и т. д.

5. Решение простейших НРУ с использованием остаточных операторов в формах (42)–(43) и (45)–(48) может иметь существенные требования к

вычислительным ресурсам, особенно для матричного представления нечетких соответствий большой размерности.

6. Решение простейших НРУ с использованием остаточных операторов в аналитической форме не всегда возможно. Поэтому, хотя реализация в этом случае более простая и не имеет существенных требований к вычислительным ресурсам, не является универсальным.

7. Результаты решения простейших НРУ с использованием программной реализации остаточных операторов в аналитической форме и в формах (42)–(43) и (45)–(48) совпадают.

Список использованных источников:

1. Асаи К. и др. Прикладные нечеткие системы: пер. с японского / Под ред. Т. Тэрано, К. Асаи, М. Сугэно. — Москва : Мир, 1993. — 386 с.

2. Блюмин С. Л. и др. Нечеткая логика: алгебраические основы и приложения: Монография / С. Л. Блюмин, И. А. Шуйкова, П. В. Сараев, И. В. Черпаков. — Липецк : ЛЭГИ. — 2002. — 111 с.

3. Кофман А., Хил Алуха Х. Введение теории нечетких множеств в управлении предприятиями: Пер. с исп. / А. Кофман, Х. Хил Алуха — Минск : Выш. шк., 1992. — 224 с.

4. Мелихов А. Н., Бернштейн Л. С., Коровин С. Я. Ситуационные советуемые системы с нечеткой логикой / А. Н. Мелихов, Л. С. Бернштейн, С. Я. Коровин. — Москва : Наука, Гл.ред. физ.-мат. лит. — 1990. — 272 с.

5. Пегат А. Нечеткое моделирование и управление / А. Пегат; под ред. Ю. В. Тюменцева. — 2-е изд. — Москва : БИНОМ. Лаборатория знаний, 2015. — 798 с.

6. Черпаков И. В. Моделирование и анализ связи показателей социально-экономических систем с использованием прямой задачи для нечетких соответствий // ЭФО: Экономика. Финансы. Общество. — 2023. — №2 (6). — С. 92–111. — DOI: 10.24412/2782-4845-2023-6-92-111.

7. Черпаков, И. В. Основы программирования : учебник и практикум для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2023. — 219 с. — (Высшее образование). — ISBN 978-5-9916-9983-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511703> (дата обращения: 17.08.2023, доступ по логину и паролю).

8. Черпаков, И. В. Теоретические основы информатики : учебник и практикум для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2023. — 353 с. — (Высшее образование). — ISBN 978-5-9916-8562-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/511750> (дата обращения: 17.08.2023, доступ по логину и паролю).

9. De Baets В. Analytic Solution Methods for Fuzzy Relational Equations // Fundamentals of Fuzzy Sets: Handbooks of Fuzzy Sets Series. — Dordrecht : Kluwer, 2000. — Vol. 1. — Ch. 6. — 50 pp.

10. De Baets B., Fodor J. Residual operators of uninorms // Soft Computing. — №3. — 1999. — P. 89–100.

11. Kandasamy V., Smarandache F. Fuzzy Relational Maps And Neutrosophic Relational Maps / V. Kandasamy, F. Smarandache — Hexis : Church Rock. — 2004. — 301 pp.

Сведения об авторе / Information about the author:

Черпаков Игорь Владимирович – доцент кафедры «Учет и информационные технологии в бизнесе» Липецкого филиала Финансового университета при Правительстве РФ, к.ф.-м.н., E-mail: ivcherpakov@fa.ru / **Cherpakov Igor Vladimirovich** - Associate Professor of the Department of Accounting and Information Technologies in Business, Lipetsk Branch of the Financial University under the Government of the Russian Federation, Cand. Sci. (Physics and Math), E-mail: ivcherpakov@fa.ru.

SPIN РИИЦ: 9294-7437

ORCID 0009-0007-5592-0145

Дата поступления статьи: 14.09.2023

Принято решение о публикации: 10.10.2023

Автор прочитал и одобрил окончательный вариант рукописи.

Конфликт интересов: автор заявляет об отсутствии конфликта интересов.