

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра Статистического Моделирования

«Научно-исследовательская работа» (семестр 8)

РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ И РЕШЕНИЕ ЗАДАЧ ПРИНЯТИЯ
РЕШЕНИЙ С ПОМОЩЬЮ МЕТОДОВ ТРОПИЧЕСКОЙ МАТЕМАТИКИ.

Выполнил:

Ткаченко Егор Андреевич
группа 19.Б04-мм

Научный руководитель:

д. ф.-м. н., профессор
Кривулин Николай Кимович

Санкт-Петербург

2023

Оглавление

Введение	3
Глава 1. Многокритериальная задача принятия решений	4
Глава 2. Элементы тропической математики	5
Глава 3. Решение многокритериальной задачи парных сравнений . . .	7
Глава 4. Разработка программных средств	9
4.1. Разработка структур для хранения чисел	9
4.1.1. Структура А	9
4.1.2. Структура В	10
4.2. Матрицы	11
4.3. Сравнение производительности структур	11
Заключение	13
Список литературы	14

Введение

Многокритериальные задачи оценки альтернатив на основе парных сравнений составляют важный класс задач принятия решений, которые встречаются во многих областях научной и практической деятельности. Пусть имеется набор альтернатив принятия некоторого решения. Известны количественные результаты парных сравнений, при которых любые две альтернативы сравниваются между собой в соответствии с несколькими критериями. Результаты сравнений могут быть получены, например, путем опроса респондентов или с помощью других процедур сравнения. Требуется на основе относительных результатов парных сравнений определить абсолютный рейтинг каждой альтернативы для принятия решения. Такие задачи встречаются при принятии управленческих решений в менеджменте, изучении предпочтений покупателей в маркетинге, анализе социологических опросов в социологии, прогнозе результатов выборов в политологии и в других областях [1].

Одним из подходов к решению является метод аппроксимации матриц парных сравнений в \log -чебышевской метрике [2, 3, 4, 5]. Этот подход позволяет получить аналитическое решение задачи в терминах \max -алгебры – одной из алгебраических систем с идемпотентными операциями, которые изучает тропическая математика [6, 7, 8].

Численное решение задач оптимизации с помощью методов тропической математики, включая решение задачи принятия решений, требует применения новых программных инструментов, предназначенных для вычислений с идемпотентными операциями. Целью настоящей работы является разработка эффективных программных средств символьных вычислений с использованием \max -алгебры.

В этом семестре была разработана и реализована вторая структура, проведены сравнения с прошлой.

Глава 1

Многокритериальная задача принятия решений

Рассмотрим задачу оценки рейтингов альтернатив на основе парных сравнений, в которой n альтернатив $\mathcal{A}_1, \dots, \mathcal{A}_n$ сравниваются попарно по m критериям. Пусть $\mathbf{A}_k = (a_{ij}^{(k)})$ обозначает матрицу порядка n , где элемент $a_{ij}^{(k)} > 0$ показывает во сколько раз альтернатива \mathcal{A}_i превосходит альтернативу \mathcal{A}_j в соответствии с критерием $k = 1, \dots, m$. Критерии также сравниваются попарно, а результаты их сравнений образуют матрицу $\mathbf{C} = (c_{kl})$, где c_{kl} показывает во сколько раз критерий k важнее для принятия решения, чем l . Необходимо на основе матриц парных сравнений \mathbf{C} и $\mathbf{A}_1, \dots, \mathbf{A}_m$ найти индивидуальный рейтинг каждой альтернативы [1].

Ниже будет рассмотрено решение задачи на основе log-чебышевской аппроксимации с применением методов тропической алгебры.

Глава 2

Элементы тропической математики

В этой главе приводятся основные обозначения и понятия тропической \max -алгебры [6, 7, 8], которые потребуются в дальнейшем.

\max -алгеброй называется множество неотрицательных вещественных чисел \mathbb{R}_+ с операциями сложения и умножения. Сложение обозначается знаком \oplus , определено как максимум: $x \oplus y = \max\{x, y\}$ и обладает свойством идемпотентности: $x \oplus x = x$ для любых $x, y \in \mathbb{R}_+$. Умножение определено и обозначается как обычно.

Векторные и матричные операции выполняются по обычным правилам с заменой арифметического сложения на операцию \oplus . Нулевой вектор обозначается символом $\mathbf{0}$. Для ненулевого вектора-столбца $\mathbf{x} = (x_j)$ определен мультипликативно сопряженный вектор-строка $\mathbf{x}^- = (x_j^-)$, где $x_j^- = x_j^{-1}$, если $x_j \neq 0$, и $x_j^- = 0$ в противном случае. Для ненулевой матрицы $\mathbf{A} = (a_{ij})$ определена мультипликативно сопряженная матрица $\mathbf{A}^- = (a_{ij}^-)$, где $a_{ij}^- = a_{ji}^{-1}$, если $a_{ji} \neq 0$, иначе $a_{ij}^- = 0$.

Вектор \mathbf{b} линейно зависит от векторов $\mathbf{a}_1, \dots, \mathbf{a}_n$, если существует набор чисел $x_1, \dots, x_n \in \mathbb{R}_+$ такой, что $\mathbf{b} = x_1 \mathbf{a}_1 \oplus \dots \oplus x_n \mathbf{a}_n$. Коллинеарность двух векторов имеет обычный смысл: векторы \mathbf{a} и \mathbf{b} являются коллинеарными, если $\mathbf{b} = x \mathbf{a}$ для некоторого $x \in \mathbb{R}_+$.

Множество всех линейных комбинаций $x_1 \mathbf{a}_1 \oplus \dots \oplus x_n \mathbf{a}_n$ образует тропическое линейное пространство. Любой вектор \mathbf{y} пространства выражается с помощью матрицы $\mathbf{A} = (\mathbf{a}_1, \dots, \mathbf{a}_n)$, составленной из этих векторов как столбцов, и вектора $\mathbf{x} = (x_1, \dots, x_n)^T$ в виде $\mathbf{y} = \mathbf{A} \mathbf{x}$.

Рассмотрим квадратные матрицы с элементами из \max -алгебры. Единичная матрица обозначается символом \mathbf{I} и имеет обычный вид. Целая неотрицательная степень квадратной матрицы \mathbf{A} определена для всех натуральных p так, что $\mathbf{A}^0 = \mathbf{I}$, $\mathbf{A}^p = \mathbf{A}^{p-1} \mathbf{A} = \mathbf{A} \mathbf{A}^{p-1}$.

След матрицы $\mathbf{A} = (a_{ij})$ порядка n вычисляется по формуле

$$\text{tr } \mathbf{A} = a_{11} \oplus \dots \oplus a_{nn}.$$

Спектральным радиусом матрицы \mathbf{A} называется число, которое вычисляется по

формуле

$$\lambda = \operatorname{tr} \mathbf{A} \oplus \cdots \oplus \operatorname{tr}^{1/n}(\mathbf{A}^n) = \bigoplus_{i=1}^n \operatorname{tr}^{1/i}(\mathbf{A}^i).$$

При условии, что $\lambda \leq 1$, определен оператор Клини (звезда Клини), который сопоставляет матрице \mathbf{A} матрицу

$$\mathbf{A}^* = \mathbf{I} \oplus \mathbf{A} \oplus \cdots \oplus \mathbf{A}^{n-1} = \bigoplus_{i=0}^{n-1} \mathbf{A}^i.$$

Глава 3

Решение многокритериальной задачи парных сравнений

Рассмотрим многокритериальную задачу парных сравнений с матрицами парных сравнений альтернатив $\mathbf{A}_1, \dots, \mathbf{A}_m$ и матрицей сравнений критериев \mathbf{C} . Приведем алгоритм решения, который использует аппроксимацию матриц парных сравнений в log-чебышевской метрике и подробнее описанный в работах [2, 3, 4].

1. Для матрицы \mathbf{C} находится спектральный радиус λ , составляется матрица $\lambda^{-1}\mathbf{C}$, а затем в параметрической форме определяется вектор весов критериев

$$\mathbf{w} = (\lambda^{-1}\mathbf{C})^*\mathbf{v}, \quad \mathbf{v} > \mathbf{0}, \quad \lambda = \bigoplus_{i=1}^m \text{tr}^{1/i}(\mathbf{C}^i).$$

2. Если вектор \mathbf{w} не единственный (с точностью до положительного множителя), то определяются наилучший и наихудший дифференцирующие векторы весов.

2.1. Наилучший дифференцирующий вектор весов имеет вид

$$\mathbf{w}_1 = \mathbf{P}(\mathbf{I} \oplus \mathbf{P}_{lk}^- \mathbf{P})\mathbf{v}_1, \quad \mathbf{v}_1 > \mathbf{0},$$

где матрица $\mathbf{P} = (\mathbf{p}_j)$ получена из $(\lambda^{-1}\mathbf{C})^*$ удалением линейно зависимых столбцов, матрица \mathbf{P}_{lk} получена из $\mathbf{P} = (p_{ij})$ обнулением всех элементов, кроме p_{lk} ,

$$k = \arg \max_j \mathbf{1}^T \mathbf{p}_j \mathbf{p}_j^- \mathbf{1}, \quad l = \arg \max_i p_{ik}^{-1}.$$

2.2. Наихудший дифференцирующий вектор весов имеет вид

$$\mathbf{w}_2 = (\Delta^{-1} \mathbf{1} \mathbf{1}^T \oplus \lambda^{-1}\mathbf{C})^*\mathbf{v}_2, \quad \mathbf{v}_2 > \mathbf{0}, \quad \Delta = \mathbf{1}^T (\lambda^{-1}\mathbf{C})^* \mathbf{1}.$$

3. С помощью векторов $\mathbf{w}_1 = (w_i^{(1)})$ и $\mathbf{w}_2 = (w_i^{(2)})$ строятся взвешенные суммы матриц парных сравнений альтернатив:

$$\mathbf{B} = \bigoplus_{i=1}^m w_i^{(1)} \mathbf{A}_i, \quad \mathbf{D} = \bigoplus_{i=1}^m w_i^{(2)} \mathbf{A}_i.$$

4. На основе матрицы \mathbf{B} вычисляется наилучший дифференцирующий вектор рейтингов альтернатив.

4.1. Для матрицы \mathbf{B} находится спектральный радиус μ , составляется матрица $\mu^{-1}\mathbf{B}$, а затем определяется вектор рейтингов

$$\mathbf{x}_1 = (\mu^{-1}\mathbf{B})^*\mathbf{u}_1, \quad \mathbf{u}_1 > \mathbf{0}, \quad \mu = \bigoplus_{i=1}^n \text{tr}^{1/i}(\mathbf{B}^i).$$

4.2. Если вектор рейтингов \mathbf{x}_1 не единственный, то строится наилучший дифференцирующий вектор

$$\mathbf{x}'_1 = \mathbf{Q}(\mathbf{I} \oplus \mathbf{Q}_{lk}^- \mathbf{Q})\mathbf{u}'_1, \quad \mathbf{u}'_1 > \mathbf{0},$$

где матрица $\mathbf{Q} = (\mathbf{q}_j)$ получена из $(\mu^{-1}\mathbf{B})^*$ удалением линейно зависимых столбцов, матрица \mathbf{Q}_{lk} получена из $\mathbf{Q} = (q_{ij})$ обнулением всех элементов, кроме q_{lk} ,

$$k = \arg \max_j \mathbf{1}^T \mathbf{q}_j \mathbf{q}_j^{-1} \mathbf{1}, \quad l = \arg \max_i q_{ik}^{-1}.$$

5. На основе матрицы \mathbf{D} вычисляется наихудший дифференцирующий вектор рейтингов альтернатив.

5.1. Для матрицы \mathbf{D} находится спектральный радиус ν , составляется матрица $\nu^{-1}\mathbf{D}$, а затем определяется вектор рейтингов

$$\mathbf{x}_2 = (\nu^{-1}\mathbf{D})^*\mathbf{u}_2, \quad \mathbf{u}_2 > \mathbf{0}, \quad \nu = \bigoplus_{i=1}^n \text{tr}^{1/i}(\mathbf{D}^i).$$

5.2. Если вектор рейтингов \mathbf{x}_2 не единственный, то строится наихудший дифференцирующий вектор

$$\mathbf{x}'_2 = (\delta^{-1}\mathbf{1}\mathbf{1}^T \oplus \nu^{-1}\mathbf{D})^*\mathbf{u}'_2, \quad \mathbf{u}'_2 > \mathbf{0}, \quad \delta = \mathbf{1}^T(\nu^{-1}\mathbf{D})^*\mathbf{1}.$$

На всех этапах алгоритма может быть получен не единственный вектор (весов, рейтингов), а набор векторов, которые будут определять некоторое пространство решений. В этом случае, важной оказывается задача сокращения числа векторов в наборе за счет удаления векторов, линейно зависимых от остальных.

Глава 4

Разработка программных средств

4.1. Разработка структур для хранения чисел

При проверке линейной зависимости векторов использование типов с плавающей точкой может привести к неточным результатам, поэтому была разработана структура для хранения чисел основанная на целочисленных типах, для которых операции являются точными.

4.1.1. Структура A

В задаче принятия решений используются матрицы парных сравнений из натуральных и обратных натуральным чисел. Для аналитического решения задачи требуется предусмотреть точное выполнение операций умножения и извлечения корня натуральной степени. Рациональных чисел $\frac{a}{b}$ не достаточно из-за операции извлечения корня. Необходимо добавить к структуре числа корень целой степени: $\left(\frac{a}{b}\right)^{1/n}$.

Такое представление чисел сужает мах-алгебру на множество чисел

$$\left\{ x = \left(\frac{a}{b}\right)^{1/n} \mid a \in \mathbb{N} \cup 0, b \in \mathbb{N}, n \in \mathbb{N} \right\}.$$

Это множество замкнуто относительно операций сложения, умножения, извлечения корня целой степени, нахождения обратного элемента и линейно упорядочено.

С такой структурой операции и отношения определяются следующим образом:

- Умножение:

$$\left(\frac{a_1}{b_1}\right)^{1/n_1} \times \left(\frac{a_2}{b_2}\right)^{1/n_2} = \left(\frac{a_1^{n_2} a_2^{n_1}}{b_1^{n_2} b_2^{n_1}}\right)^{1/n_1 n_2}.$$

- Сравнение:

$$\begin{aligned} \left(\frac{a_1}{b_1}\right)^{1/n_1} < \left(\frac{a_2}{b_2}\right)^{1/n_2} &\Leftrightarrow \left(\frac{a_1^{n_2}}{b_1^{n_2}}\right)^{1/n_1 n_2} < \left(\frac{a_2^{n_1}}{b_2^{n_1}}\right)^{1/n_1 n_2} \Leftrightarrow \\ &\Leftrightarrow \frac{a_1^{n_2}}{b_1^{n_2}} < \frac{a_2^{n_1}}{b_2^{n_1}} \Leftrightarrow a_1^{n_2} b_2^{n_1} < a_2^{n_1} b_1^{n_2}. \end{aligned}$$

- Обратный элемент относительно умножения:

$$\left(\left(\frac{a}{b}\right)^{1/n}\right)^{-1} = \left(\frac{b}{a}\right)^{1/n}, \quad a \neq 0.$$

Однако, если использовать приведенные формулы, числа будут увеличиваться очень быстро. Причем, часто n_1 и n_2 оказываются равными. Это мотивирует использовать НОД в формулах:

$$n_1 = n_1^* \cdot \gcd(n_1, n_2), \quad n_2 = n_2^* \cdot \gcd(n_1, n_2).$$

- Умножение:

$$\left(\frac{a_1}{b_1}\right)^{1/n_1} \times \left(\frac{a_2}{b_2}\right)^{1/n_2} = \left(\frac{a_1^{n_2^*} a_2^{n_1^*}}{b_1^{n_2^*} b_2^{n_1^*}}\right)^{1/n_1^* \cdot \gcd(n_1, n_2) \cdot n_2^*}.$$

После умножения числитель и знаменатель сокращаются на их НОД.

- Сравнение:

$$\left(\frac{a_1}{b_1}\right)^{1/n_1} < \left(\frac{a_2}{b_2}\right)^{1/n_2} \Leftrightarrow a_1^{n_2^*} b_2^{n_1^*} < a_2^{n_1^*} b_1^{n_2^*}.$$

4.1.2. Структура В

Описанная ранее структура имеет недостаток — числитель и знаменатель становятся большими очень быстро с увеличением размера матриц. Из-за этого операция умножения замедляется, т.к. числа не помещаются в стандартные типы, для которых оптимизированы процессоры.

Идея второй структуры заключается в том, чтобы заменить умножение сложением, как это делает логарифмирование.

$$\log ab = \log a + \log b.$$

Структура В представляет числа в виде, похожем на факторизацию, но степени простых рациональны:

$$x = p_1^{a_1} p_2^{a_2} \dots p_k^{a_k}, \quad \text{где } p_i \text{ — простые, } a_i \in \mathbb{Q}.$$

Структура реализуется вектором пар натуральных чисел и рациональных с отдельным состоянием для 0.

- Умножение реализуется слиянием векторов множителей. Когда простой делитель присутствует в обоих множителях, его степени складываются и если сумма не равна нулю, он добавляется в результат. Если простой делитель присутствует только в одном из множителей, он со своей степенью добавляется в результат.

Например:

$$2^3 3^{-2} \times 3^2 5^{-1} = 2^3 3^{-2+2} 5^{-1} = 2^3 5^{-1}.$$

- Обратный по умножению получается изменением знака степеней простых.

$$x^{-1} = (p_1^{a_1} p_2^{a_2} \dots p_k^{a_k})^{-1} = p_1^{-a_1} p_2^{-a_2} \dots p_k^{-a_k}.$$

- Сравнение двух чисел можно свести к сравнению с единицей с помощью умножения одного аргумента на обратный по умножению второго аргумента (деления).

Чтобы сравнить число с единицей, все степени простых умножаются на наименьший общий множитель l знаменателей степеней a_i , чтобы степени стали целыми. Далее находится два произведения — простых (возведенных в соответствующую степень) с положительными степенями и простых с отрицательными. Два произведения сравниваются как целые числа и определяют результат исходного сравнения.

$$\begin{aligned} a < b &\Leftrightarrow a/b = p_1^{a_1} p_2^{a_2} \dots p_k^{a_k} < 1 \Leftrightarrow p_1^{la_1} p_2^{la_2} \dots p_k^{la_k} < 1 \Leftrightarrow \\ &\Leftrightarrow \prod_{i \in \{i | a_i > 0\}} p_i^{la_i} < \prod_{j \in \{j | a_j < 0\}} p_j^{-la_j}. \end{aligned}$$

Эта процедура медленная и зачастую избыточно точная, поэтому реализована более быстрая, но приближенная процедура сравнения.

Для отношения сравниваемых чисел вычисляется приближенное значение в типе с плавающей точкой и если оно достаточно далеко от 1 (разница больше 10^{-15}), то используется результат сравнения приближения с 1, иначе запускается медленный и точный метод.

4.2. Матрицы

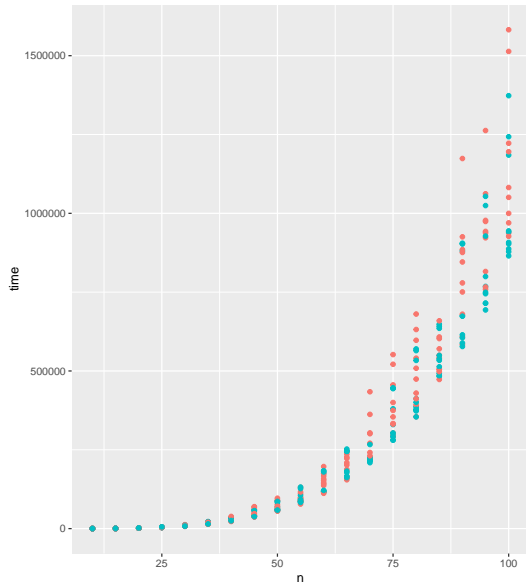
Было реализовано расширение библиотеки Eigen, позволяющее использовать описанные раньше структуры в матрицах.

4.3. Сравнение производительности структур

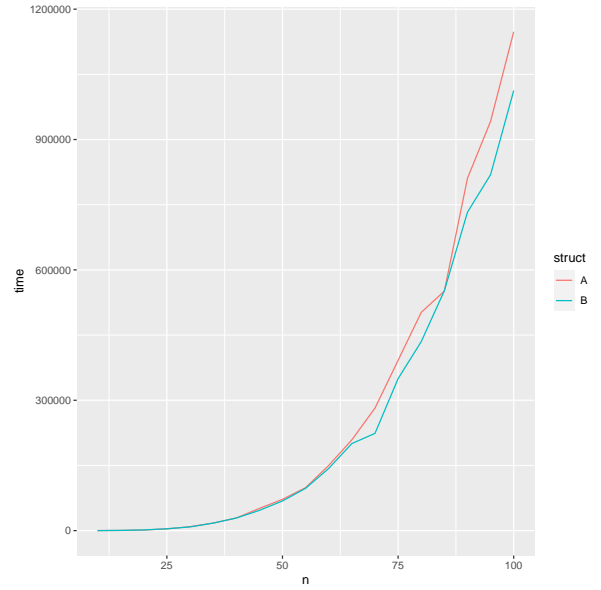
Структуры сравнивались по времени вычисления $(\lambda^{-1} \mathbf{A})^*$, где λ — спектральный радиус матрицы \mathbf{A} , \mathbf{A} — случайно сгенерированная матрица парных сравнений $n \times n$.

Асимптотика такого теста — $O(n^4(t_{\times} + t_{\oplus}))$, где t_{\times}, t_{\oplus} — сложность (время) умножения и сложения чисел, соответственно.

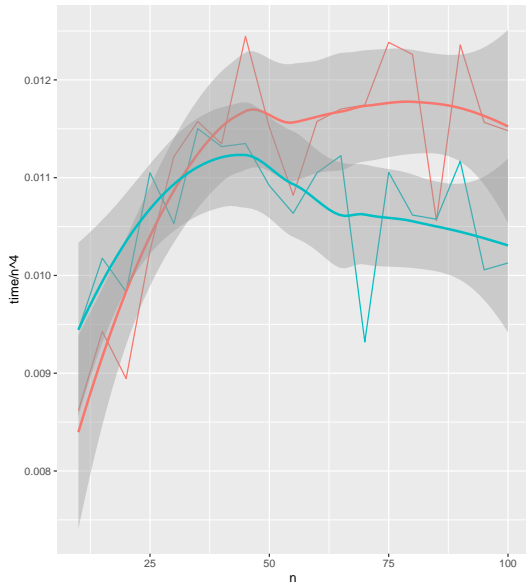
Для каждого значения n проведено по 10 тестов и найдено среднее время вычисления в миллисекундах.



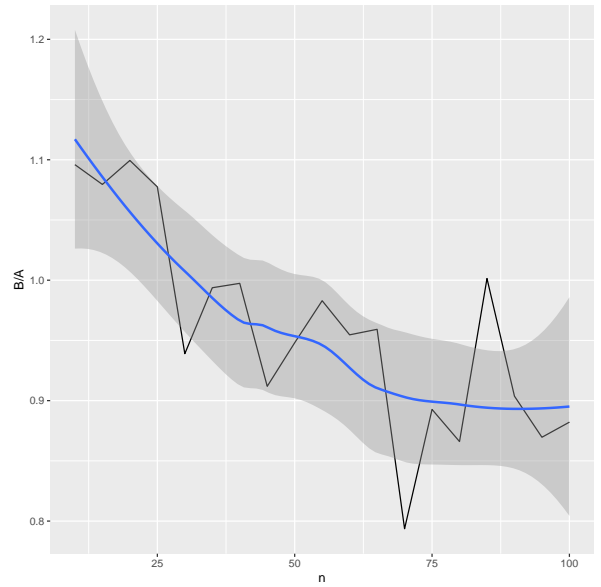
(а) Все тесты



(б) Среднее тестов



(в) Масштабирование по размеру матриц



(г) Отношение времени вычисления В и А

Рис. 4.1. Результаты тестирования.

Как видно из графика 4.1(г), при маленьких размерах матрицы структура А быстрее, а начиная примерно с $n = 30$, структура В становится быстрее. При размерах матрицы $n = 100$ (время вычисления порядка 20 минут) структура В лучше на 10%.

Заключение

Для решения многокритериальных задач парных сравнений разработаны модели представления данных, алгоритмы точных вычислений, их программная реализация и проведено сравнение.

Полученные результаты могут оказаться полезными для решения других задач, где требуется обеспечить точные вычисления, например для задач криптографии.

Список литературы

1. Саати Т. Принятие решений. Метод анализа иерархий / пер. с англ. Р. Г. Вачнадзе. М.: Радио и связь. — 1993.
2. Кривулин Н. К., Агеев В. А. Методы тропической оптимизации в многокритериальных задачах оценки альтернатив на основе парных сравнений // Вестн. С.-Петерб. ун-та // Прикладная математика. — 2019. — Т. 15. — С. 472–488.
3. Krivulin N., Sergeev S. Tropical implementation of the Analytical Hierarchy Process decision method // Fuzzy Sets and Systems. — 2019.
4. Krivulin Nikolai, Prinkov Alexey, Gladkikh Igor. Using Pairwise Comparisons to Determine Consumer Preferences in Hotel Selection // Mathematics. — 2022. — Vol. 10, no. 5. — P. 1–25.
5. Кривулин Н. К., Абильдаев Т., Горшечникова В. Д., Капаца Д. и Магдич Е. А. // Мандрикова А. А. О решении многокритериальных задач принятия решений на основе парных сравнений // Компьютерные инструменты в образовании. — № 2. 2020. — С. 27–58.
6. Маслов В. П., Колокольников В. Н. Идемпотентный анализ и его применение в оптимальном управлении. — 1994.
7. Butkovič P. Max-linear Systems: Theory and Algorithms. — 2010.
8. Heidergott B., Olsder G., Woude J. Van Der. — 2006.