

# ПРОГРАММНЫЙ КОМПЛЕКС НАХОЖДЕНИЯ РЕЗУЛЬТАТОВ ОПЕРАЦИЙ НАД ЭКСТРЕМАЛЬНЫМИ ГИПЕРГРАФАМИ

## SOFTWARE PACKAGE FOR FINDING THE RESULTS OF OPERATIONS ON EXTREME HYPERGRAPHS

**I. Beretskiy  
I. Irbitskiy  
E. Egorova  
A. Mokryakov  
T. Chernova**

*Summary.* This paper presents the results of implementing a software package that performs a number of Boolean operations on  $k$ -homogeneous extreme hypergraphs. In particular, it allows you to find the database of the resulting complex. The base is a shorter and more understandable form of writing an extreme hypergraph. Since only one database uniquely corresponds to a single hypergraph, you can work with them without any danger of misinterpreting the data. The presented software package allows you to save time that scientists working with hypergraphs will spend on calculations. This will allow them to focus on their tasks and speed up the implementation of their research.

*Keywords:* hypergraph, logic algebra, base, complex, extreme hypergraph,  $k$ -homogeneous hypergraph, C#, windows forms.

**Берецкий Игорь Сергеевич**

Московский авиационный институт (национальный  
исследовательский университет)  
ISberetskij@mai.ru

**Ирбитский Илья Сергеевич**

Московский авиационный институт (национальный  
исследовательский университет)  
ILSirbitskij@mai.ru

**Егорова Евгения Кирилловна**

К.ф.-м.н., доцент, Московский авиационный  
институт (национальный исследовательский  
университет)  
egorovaek@mat.i.ru

**Мокряков Алексей Викторович**

К.ф.-м.н., доцент, Московский авиационный  
институт (национальный исследовательский  
университет); доцент, Российский государственный  
университет имени А. Н. Косыгина  
MokryakovAlVik@gmail.com

**Чернова Татьяна Александровна**

Д.т.н., профессор, Московский авиационный  
институт (национальный исследовательский  
университет)  
Chernovata@mai.ru

*Аннотация.* В работе представлены результаты реализации программного комплекса, выполняющего ряд булевых операций над  $k$ -однородными экстремальными гиперграфами. В том числе он позволяет находить базу результирующего комплекса. База — это более краткая и понятная форма записи экстремального гиперграфа. Так как одному гиперграфу однозначно соответствует только одна база, с ними можно работать без какой-либо опасности неправильной интерпретации данных. Представленный программный комплекс позволяет сэкономить время, которое учёные, работающие с гиперграфами, будут тратить на вычисления. Это позволит им сконцентрироваться на своих задачах и ускорить реализацию их исследований.

*Ключевые слова:* гиперграф, алгебра логики, база, комплекс, экстремальный гиперграф,  $k$ -однородный гиперграф, c#, windows forms.

## Введение

**Т**еория гиперграфов применяется в различных областях человеческой деятельности: химии [1], логистике [2], хранении данных в реляционных БД [3], производственных процессах [4], математической оптимизации [5], криптографии [6], программировании [7], распределении ресурсов [8].

Для начала определимся с основными понятиями.

*Гиперграф* — это обобщение графа, в котором каждым ребром могут соединяться не только две вершины, но и любые подмножества вершин [9].

Для обозначения количества вершин, которое соединяет ребро графа, введено понятие *однородности*. Од-

нородность означает то, что у рассматриваемых гиперграфов размерность всех рёбер одинакова. Например, граф без петель это 2-однородный гиперграф, у которого каждое ребро соединяет 2 вершины. У 3-однородного гиперграфа каждое ребро соединяет 3 вершины, и так далее.

В данной работе исследуются операции над совершенными и экстремальными гиперграфами. *Совершенство* проявляется тогда, когда не существует неизоморфного гиперграфа с тем же вектором степеней, как у рассматриваемого графа. *Экстремальность* [10] проявляется тогда, когда выполняется совершенство и вектор степеней вершин гиперграфа упорядочен по не возрастанию.

Для определения совершенности и экстремальности гиперграфов мы будем рассматривать их вектора с помощью алгоритма Мокрякова [11]. Для этого добавим несколько понятий.

Пусть вектор  $A$  из  $\mathbb{Z}_+^n$  есть реализуемый в граф вектор и  $\{G(A)\}$  — множество его реализаций.

Вектор  $A$  из  $\mathbb{Z}_+^n$ , где  $n \geq 2$ , называется *совершенным*, если  $|\{G(A)\}|=1$ . При этом единственная реализация  $G(A)$  называется *совершенным графом*.

Вектор  $A$  из  $\overline{\mathbb{Z}}_+^n$ , где  $n \geq 2$ , называется *экстремальным*, если  $A$  — совершенный вектор. При этом единственная реализация  $G(A)$  называется *экстремальным графом*.

У каждого экстремального гиперграфа существует *база* [12]. Перед определением этого понятия введём несколько других обозначений.

Для множества вершин  $U(n)$ , где  $n \geq 2$ , и  $k \in \mathbb{Z}, 0 \leq k \leq n-1$ , через

$$S^{k+1}(n) = \{\{u_{i_1}, \dots, u_{i_{k+1}}\} : u_{i_1} \in U(n), u_{i_p} \neq u_{i_q} \text{ при } p \neq q\}$$

обозначим множество всех  $(k+1)$ -элементных подмножеств из  $U(n)$ . Пара множеств  $\{U(n), S^{k+1}\}$ , где  $S^{k+1} \subseteq S^{k+1}$ , которую обозначим

$$H^k = H^k(V(n), E^k = E^k(H^k)),$$

называется *k-комплексом* [13].

Рассмотрим множество всех кубических симметричных бинарных матриц смежности  $n$ -вершинных 2-комплексов  $\{X_n^{(3)} = (x_{ijk}) : 1 \leq i, j, k \leq n\}$  [14].

На множестве всех элементов, состоящих из трёх попарно различных индексов

$$I_n^3 = \{\{i, j, k\} : 1 \leq i < j < k \leq n\},$$

введено отношение частичного порядка, задающее частичный порядок для 2-мерных симплексов из  $S^3(n)$  и любого множества симплексов из  $S_0^3 \subseteq S^3(n)$ . Элементы из  $I_n^3$  называются *тройками*.

На элементах множества  $I_n^3$  введём отношение частичного порядка: положим  $\{i, j, k\} \geq \{p, q, l\}$ , если  $i \geq p, j \geq q, k \geq l$  и  $\{i, j, k\} > \{p, q, l\}$  при  $\{i, j, k\} \geq \{p, q, l\}$  и  $\{i, j, k\} \neq \{p, q, l\}$ .

Пусть  $G^2$  произвольный  $n$ -вершинный 2-комплекс с матрицей смежности

$$X_n^{(3)}(G^2) = (x_{ijk}).$$

Будем применять обозначение

$$G^2 \cong (x_{ijk}) = X_n^{(3)}.$$

Для множества троек, соответствующих симплексам комплекса  $G^2$ , введём обозначение

$$I_n^3(G^2) = \{\{i, j, k\} \in I_n^3 : x_{ijk} = 1\}.$$

После ввода всех этих терминов перейдём к самому понятию базы.

Пусть

$$G^2 = G^2(U(n), S^3(G^2)) \cong (x_{ijk})$$

произвольный 2-комплекс, где

$$S^3(G^2) \neq \emptyset.$$

Множество

$$\overline{I}_n^3(G^2) = \{\{i, j, k\}\}$$

называется *базой* для комплекса  $G^2$  при справедливости условий:

а) Если  $\{p, q, l\} \in \overline{I}_n^3(G^2)$ , то  $\{p, q, l\} \in I_n^3(G^2)$ , то есть  $\overline{I}_n^3(G^2) \subseteq I_n^3(G^2)$ ;

б) Если  $\{p_1, q_1, l_1\}, \{p_2, q_2, l_2\} \in \overline{I}_n^3(G^2)$ , то отношение частичного порядка для указанных троек не определено;

в) Для  $\{i, j, k\} \in \overline{I}_n^3(G^2) \setminus I_n^3(G^2)$  существует тройка  $\{p, q, l\} \in \overline{I}_n^3(G^2)$  такая, что  $\{i, j, k\} < \{p, q, l\}$ ;

д) Если  $\{i, j, k\} < \{p, q, l\}$ , где  $\{p, q, l\} \in \overline{I}_n^3(G^2)$ , то  $\{i, j, k\} \in I_n^3(G^2)$ .

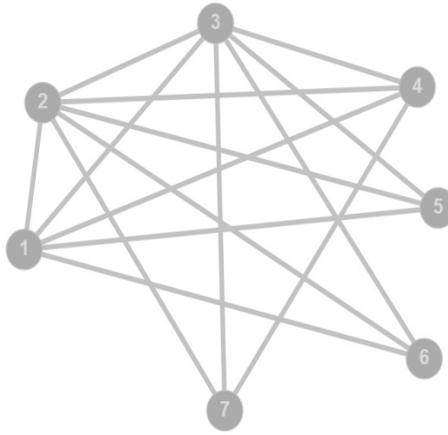


Рис. 1. Гиперграф  $G_1$

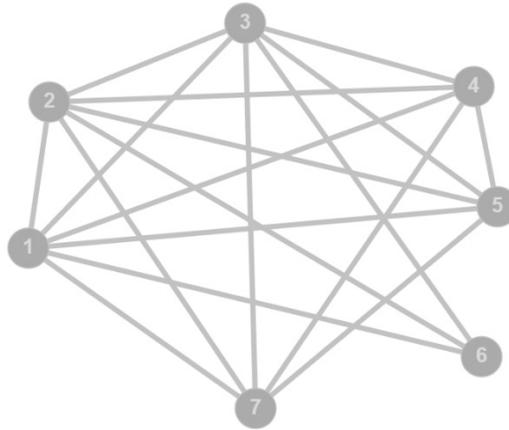


Рис. 2. Гиперграф  $G_2$

После ввода основных понятий посмотрим на примерах, как они реализуются в 2-однородных гиперграфах.

**Примеры гиперграфов**

**Пример 1**

Рассмотрим 2-однородный гиперграф  $G_1$ :

Его вектор имеет вид  $(5,6,6,4,3,3,3)$ . Так как степени вершин не упорядочены по не возрастанию он не является экстремальным.

Проверим его на совершенности с помощью алгоритма Хакими [14]:

1. Упорядочим степени по не возрастанию:  $(6,6,5,4,3,3,3)$ .
2. Построим редуccionные вектора:

$$A \quad 6 \ 5 \ 4 \ 3 \ 3 \ 3$$

$$A' \ 5 \ 4 \ 3 \ 2 \ 2 \ 2$$

$$A'' \ 3 \ 2 \ 1 \ 1 \ 1$$

Уже на втором редуccionном шаге вектор перестаёт быть строго приводимым:

$3 \neq 5 - 1 - 0$ . Из этого следует вывод, что данный вектор не является совершенным, как и соответствующий ему граф.

В итоге граф  $G_1$  является графом общего вида.

**Пример 2**

Рассмотрим 2-однородный гиперграф  $G_2$ :

Его вектор имеет вид  $(6,6,6,5,5,3,5)$ . Так как степени вершин не упорядочены по не возрастанию он не является экстремальным.

Проверим его на совершенности с помощью алгоритма Хакими:

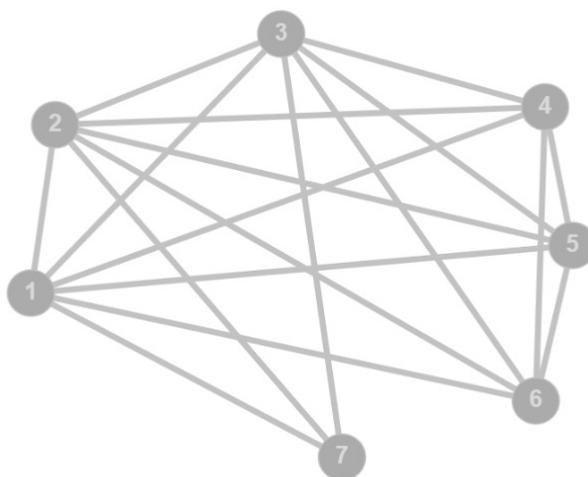


Рис. 3. Гиперграф  $G_3$

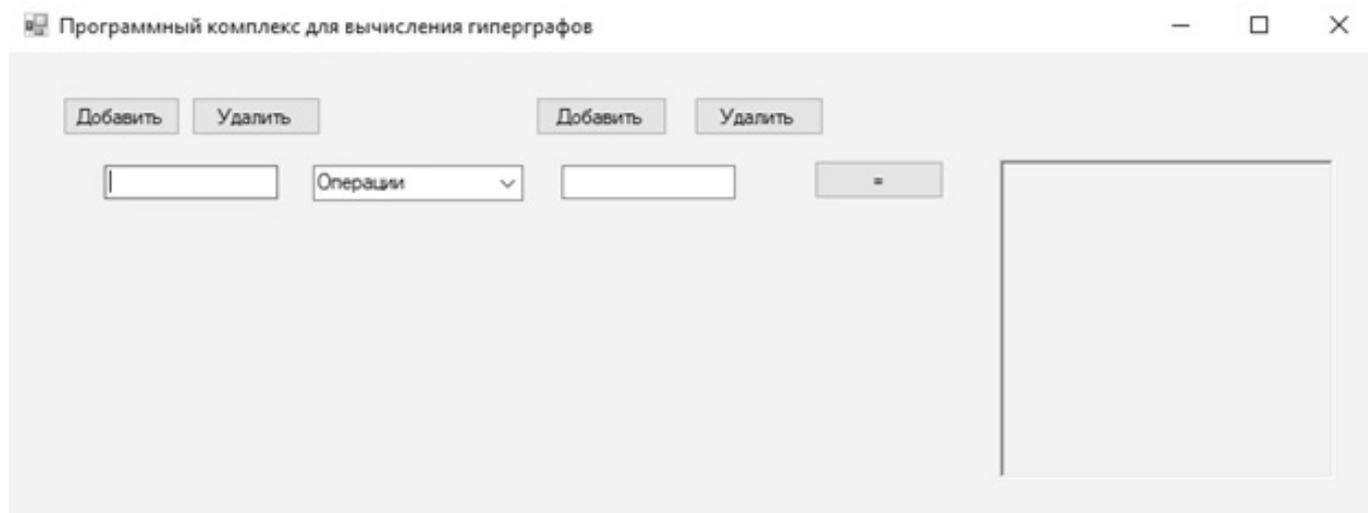


Рис. 4. Стартовый экран

3. Упорядочим степени по не возрастанию:  
(6,6,6,5,5,5,3).

4. Построим редуционные вектора:

$A$  6 6 6 5 5 5 3  
 $A'$  5 5 4 4 4 2  
 $A''$  4 3 3 3 1  
 $A'''$  2 2 2 0  
 $A^{(4)}$  1 1 0

Мы дошли до самого конца, т. Е. каждый из векторов был строго приводимым. Из этого следует вывод, что данный вектор является совершенным, как и соответствующий ему граф.

В итоге граф  $G_2$  является графом совершенным, но не экстремальным.

**Пример 3**

Рассмотрим 2-однородный гиперграф  $G_3$ :

Данный граф является почти полной копией графа из 2го примера,  $G_2$ . Единственным отличием является перестановка вершин 6 и 7. Теперь вектор вершин имеет вид (6,6,6,5,5,5,3) и с самого начала упорядочен по не возрастанию. Мы уже проверили, что этот вектор является совершенным, что вкупе с изначальной упорядоченностью даёт ему свойство экстремальности.

В итоге граф  $G_3$  является экстремальным.

Так как граф  $G_3$  является экстремальным, у него существует база. Для её нахождения выпишем все рёбра, которые имеются у графа:

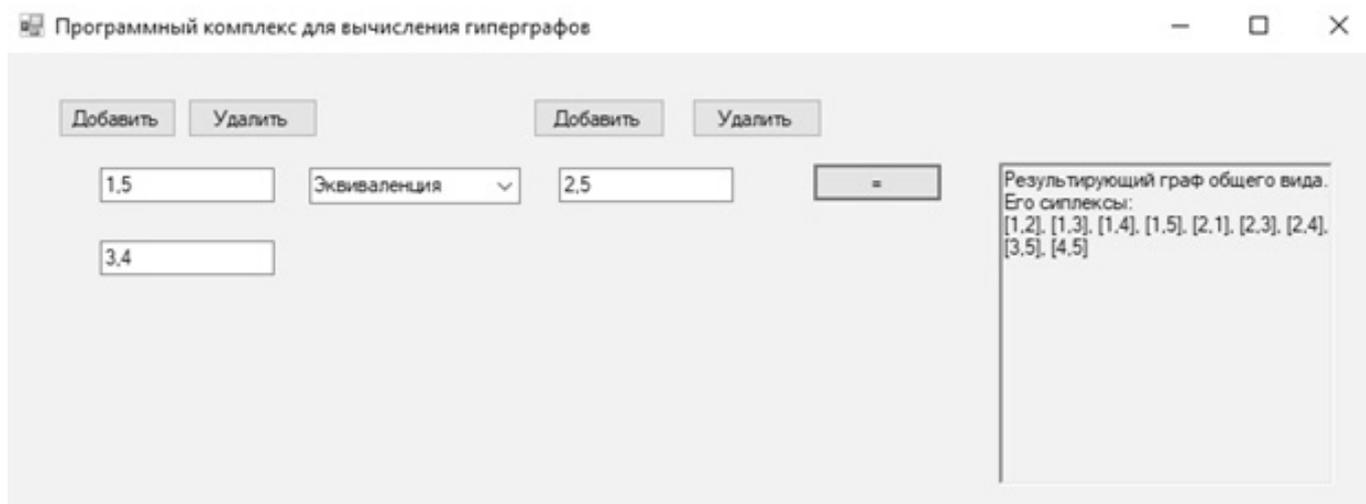


Рис. 5. Эквивалентность 2-однородных гиперграфов

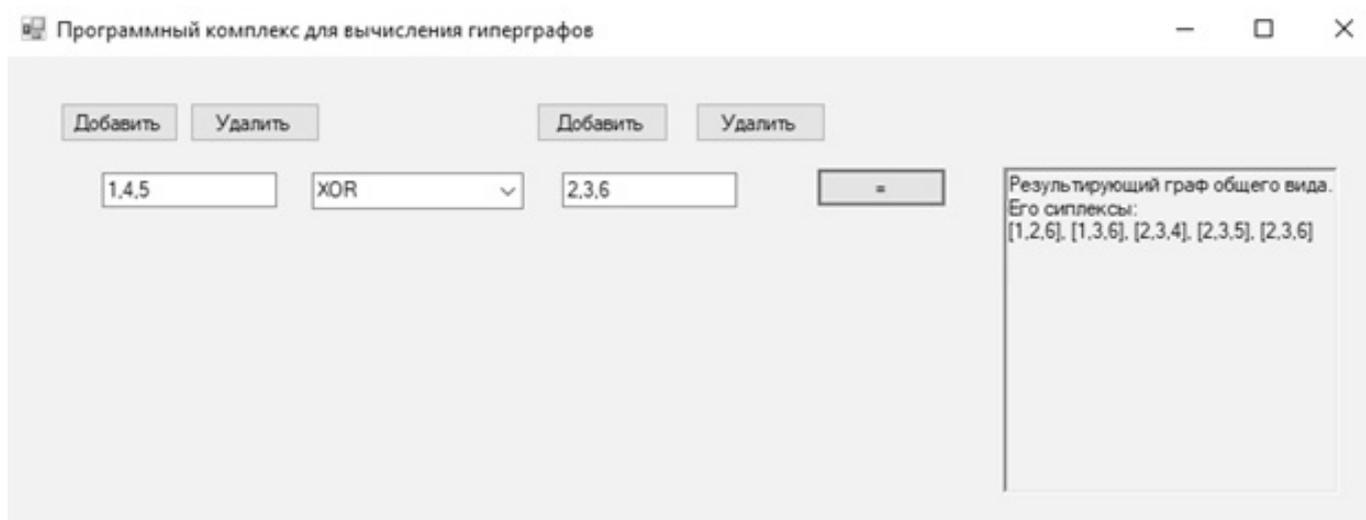


Рис. 6. XOR3-однородных гиперграфов

$\{\{1,2\}, \{1,3\}, \{1,4\}, \{1,5\}, \{1,6\}, \{1,7\}, \{2,3\}, \{2,4\}, \{2,5\}, \{2,6\}, \{2,7\}, \{3,4\}, \{3,5\}, \{3,6\}, \{3,7\}, \{4,5\}, \{4,6\}, \{5,6\}\}$ .

По определению базы её легко найти: рёбра  $\{1,2\}$  —  $\{3,7\}$  “сворачиваются” в  $\{3,7\}$ ,  $\{4,5\}$  —  $\{5,6\}$  в  $\{5,6\}$ . В итоге базой графа будет

$$\bar{I}_7^2 = \{\{3,7\}, \{5,6\}\}.$$

#### Вычислительный комплекс

Для нахождения баз экстремальных  $k$ -однородных гиперграфов и результатов операций над ними [15] реализован вычислительный комплекс [16].

Как примеры работы программы рассмотрим вычисления для 2-однородных, 3-однородных и 5-однородных гиперграфов. В программе можно также проводить вычисления над гиперграфами большей степени однородности, но для примера я выбрал именно эти цифры, так как благодаря меньшей степени вычисления можно проверить самому, используя таблицы истинности из теоретической части выше.

Сначала рассмотрим 2-однородные гиперграфы с базами  $\{\{1,5\}, \{3,4\}\}, \{2,5\}$  и проведём операцию эквивалентности. Данный пример отличается от дальнейших тем, что у первый гиперграф представлен двусоставной базой, состоящей из двух симплексов. В результате вычис-

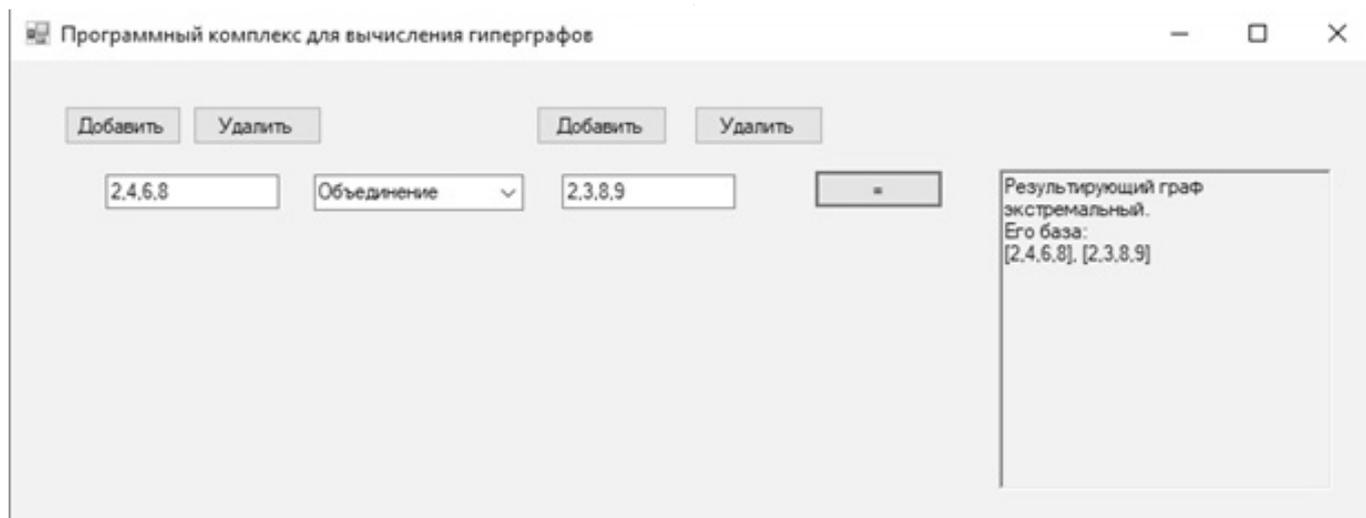


Рис. 7. Объединение 4-однородных гиперграфов



Рис. 8. Объединение 3-однородных гиперграфов

лений получим гиперграф общего вида с рёбрами  $\{[1,2], [1,3], [1,4], [1,5], [2,1], [2,3], [2,4], [3,5], [4,5]\}$ .

Теперь перейдём к менее тривиальному случаю, 3-однородному гиперграфу. Для примера возьмём гиперграфы с базами  $\{1,4,5\}$  и  $\{2,3,6\}$ , после чего проведём операцию XOR. У этих графов разное число вершин, но программа достроила первый граф до размеров второго, добавив к нему недостающие элементы. В результате получился гиперграф с рёбрами  $\{\{1,2,6\}, \{1,3,6\}, \{2,3,4\}, \{2,3,5\}, \{2,3,6\}\}$ .

Рассмотрим два случая, которые могут произойти при вычислении операции объединения. Сначала возьмём 4-однородные гиперграфы с базами  $\{2,4,6,8\}$

и  $\{2,3,8,9\}$ . Так как результат их объединения нельзя записать в виде одной базы, программа разбила её на две составляющих:  $\{2,4,6,8\}$  и  $\{2,3,8,9\}$ .

Второй случай более тривиальный — рассмотрим 3-однородные гиперграфы с базами  $\{1,3,4\}$  и  $\{1,2,3\}, \{1,3,10\}$ . В отличие от прошлого случая здесь возможно записать результат операции объединения в виде одного симплекса. В итоге получится база  $\{1,3,10\}$ .

Переходим к самому нетривиальному случаю, 5-однородному гиперграфу. Для примера возьмём гиперграфы с базами  $\{3,5,9,10,11\}$  и  $\{2,6,8,10,12\}$ , после чего проведём операцию пересечения. Изначально я хотел привести в дипломе результат операции

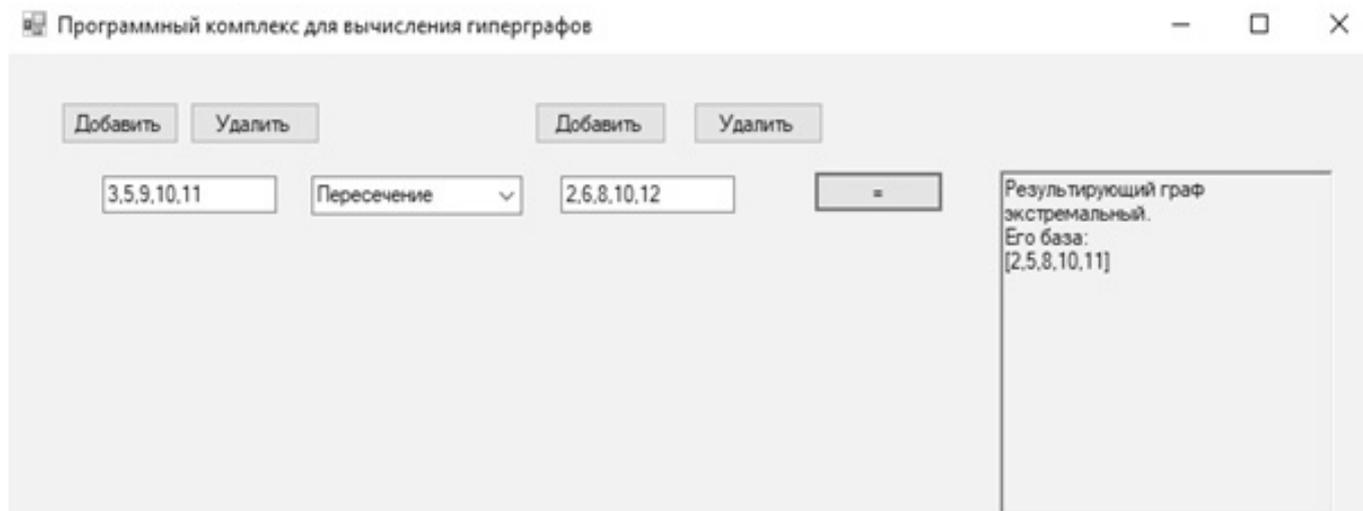


Рис. 9. Пересечение 5-однородных гиперграфов



Рис. 10. Штрих Шеффера для 2-однородных гиперграфов

Штрих Шеффера, но количество выводимых на экран симплексов оказалось слишком велико, что не позволяло оценить работу вычислительного комплекса. Поэтому я решил привести в виде примера пересечение, так как результатом этой операции является база. В результате получился экстремальный гиперграф с базой {2,5,8,10,11}.

Для показа результата работы операции Штрих Шеффера возьмём 2-однородные гиперграфы с базами {1,7} и {2,5}, {3,4}. В результате получится совершенный граф с большим количеством симплексов — именно поэтому для наглядной демонстрации я выбрал именно 2-однородные гиперграфы, т.к. при больших размерностях результат выполнения будет слишком объёмным для оценки.

#### Описание работы вычислительного комплекса

Вычислительный комплекс написан с использованием технологии Windows Forms.

Программа состоит из 3х частей: класса Application, который отвечает на отображение интерфейса, класса graf, в котором выполняются все вычисления, и вспомогательного класса CFG, перегружающего сортировку.

В классе Application хранится 4 переменные, в которых хранятся графы после ввода, а также количество полей для ввода симплексов. В нём реализованы обработчики нажатия кнопок основной формы: button1\_click

добавляет поле ввода симплекса для первого гиперграфа, `button2_click` делает то же самое для второго, процедура `button3_click` обрабатывает скрытие с формы поле ввода симплекса первого гиперграфа, `button4_click` делает то же самое для второго.

В процедуре `Application_Load` производятся необходимые подготовительные вычисления при запуске формы приложения. Основные операции проводятся в процедуре `button5_click`.

Перед тем, как я перейду к более подробному рассмотрению происходящего в основной процедуре `button5_click` нужно описать класс `graf`, который хранит в себе информацию о гиперграфах и операции над ними.

В качестве хранимых переменных класс содержит представление графа в виде строк симплексов, представление графа в матричном виде, и число вершин.

В функции `toSimplex` происходит раскрытие “скрытых” в базе рёбер гиперграфа. Например, база  $\{2,3\}$  раскроется в рёбра  $\{\{1,2\}, \{1,3\}, \{2,3\}\}$ . Процедура `toMatrix` приводит гиперграф в матричный вид. Далее класс содержит перегрузку операторов  $+$ ,  $*$ ,  $\wedge$ ,  $|$ ,  $/$ , благодаря которым можно выполнять операции объединения, пересечения, эквиваленции, XOR и Штриха Шеффера соответственно.

Затем идёт булева функция `isExtr`, которая возвращает `true` или `false` в зависимости от того, является данный гиперграф экстремальным или нет. После неё идёт ещё одна булева функция, `isSov`. Она проверяет данный гиперграф на совершенность, после чего в зависимости от результата возвращает `true` или `false`.

Затем идут “обратные” функции. Они нужны для обработки полученного после выполнения операций гиперграфа. Функция `matrixToSimplex` переводит гиперграф из матричного вида в вид строк-симплексов-рёбер. Функция `matrixToBase` запускается если искомый граф является экстремальным, она позволяет из списка симплексов-рёбер выделить базу.

На этом описание класса `graf` завершено. Также в программе есть `Зй`, вспомогательный класс `GFG`. Он насле-

дуется от интерфейса `IComparer` и нужен для перегрузки сравнения. Дело в том, что в процессе определения совершенности нужно перемещать столбцы гиперграфов в матричном виде. Это встроенная сортировка сделать не может. Поэтому нужно было перегрузить её под свои нужды.

Вернёмся к рассмотрению процедуры `button5_Click`, в которой выполняется основное тело программы.

Сначала идёт подготовка формы вывода, очищается `richTextBox1` с помощью метода `Clear`. Далее происходит перенос из симплексов гиперграфов из `textBox` полей в списки `List`. Затем введённые гиперграфы переводятся в матричные формы методом `toMatrix`, внутри которого вызывается метод `toSimplex`. После этого в зависимости от выбранной на форме операции вызывается то или иное перегруженное арифметическое действие. Затем полученный гиперграф переводится в симплексную форму процедурой `matrixToSimplex`. Далее идёт проверка на экстремальность с помощью функции `isExtr`. Если результирующий гиперграф экстремальный, вызывается процедура `matrixToBase`, которая записывает во временную строковую переменную базу полученного гиперграфа. Если гиперграф не является экстремальным, запускается функция `isSov`, по результату которой гиперграф объявляется совершенным или общего вида. В таком случае во временную переменную записывается не база гиперграфа, а его симплексное представление.

После вышеперечисленных операций на выход подаётся тип графа и его представление в виде базы или симплексов. Процедура завершает работу и управление переходит на форму, где можно ввести новые данные.

## Заключение

В настоящей работе был описан и реализован вычислительный комплекс для нахождения экстремальности гиперграфов. Кроме того, были продемонстрированы результаты работы реализованного вычислительного комплекса и описан принцип его работы. В дальнейшем предполагается развить данный программный комплекс в полнофункциональную систему работы с гиперграфами.

## ЛИТЕРАТУРА

1. Скворцова М.И., Фасхутдинова И. И., Михайлова Н. А. Гиперграфовые модели молекул углеводородов и их применение в компьютерной химии // Вестник МИТХТ им. М. В. Ломоносова. 2014. № 5. С. 86–93.
2. Капранов С.А., Кочнев А. И. Применение эвристических методов оптимизации дорожных потоков с использованием гиперграфов для определения надёжности и отказоустойчивости // Информационные системы и технологии ИСТ-2017. Материалы докладов XXIII Международной научно-технической конференции. 2017. С. 505–509.

3. Мокрозуб В.Г., Немтинов В. А., Егоров С. Я., Морозов С. В. Применение гиперграфов и реляционной базы данных для описания структуры радиотехнических систем // Успехи современной радиоэлектроники. 2009. № 3. С. 37–41.
4. Тарасов В.А., Круглов П. В., Болотина И. А. Метод формирования совокупности допустимых вариантов сборки изделий на основе применения ориентированных гиперграфов // Наука и образование: научное издание МГТУ им. Н. Э. Баумана. 2012. № 2. С. 4.
5. Миронов А.А., Мокряков А. В. Двумерные комплексы полностью описываемые степенями вершин // Труды Института системного анализа Российской академии наук. 2006. № 10. С. 178.
6. Egorova E.K., Mokryakov A.V., Suvorova A. A. The Concept of Data Encryption Using Extreme Uniform Hypergraphs // Abstracts 18th International Conference "Aviation and Cosmonautics — 2019". 2019. P. 409.
7. Головкин Ю.Б., Гусаренко А. С. Применение нечётких гиперграфов в моделях генерации WEB-компонентов // Известия высших учебных заведений. Приборостроение. 2014. № 9. С. 47–53
8. Kostyanoi D.S., Mokryakov A. V., Tsurkov V. I. Hypergraph Recovery Algorithms from a Given Vector of Vertex Degrees // Journal of Computer and Systems Sciences International. 2014. T. 53. No 4. P. 511–516.
9. Egorova E.K., Mokryakov A. V., Vang L. Development of Hypergraph Theory // J.~Computer and Systems Sciences International. 2018. V. 57. P. 109–114.
10. Mokryakov A.V., Tsurkov V. I. Reconstructing 2-complexes by a nonnegative integer-valued vector // Automation and Remote Control. 2011. T. 72. No 12. P. 2541–2552.
11. Гурченков А.А., Костяной Д. С., Мокряков А. В. Редукционные методы восстановления некоторого класса гиперграфов // Инженерный журнал: наука и инновации. 2014. № 6 (30). С. 1.
12. Mokryakov A. V. Hypergraphs as Algebraic Structures // Journal of Computer and Systems Sciences International. 2011. T. 50. No 5. P. 734–740.
13. Mironov A.A., Mokryakov A. V., Sokolov A. A. About Realization of Integer Non-Negative Numbers Tuple Into 2-Dimensional Complexes // Applied and Computational Mathematics. 2007. T. 6. No 1. P. 58–68.
14. Мокряков А.В., Селин П. С., Цурков В. И. Минимакс и восстановление по вектору в графах. М.: Физматлит, 2017. 309 с.
15. Егорова Е.К., Есенков А. С., Мокряков А. В. Операции над  $k$ -однородными гиперграфами и их векторы степеней вершин // Известия РАН. Теория и системы управления. 2020. № 3. С.75–80.
16. Берецкий И.С., Хмелевский И. Д. Замкнутость операций над классом экстремальных однородных гиперграфов // Сборник тезисов докладов ММНК Гагаринские чтения — 2020. 2020. С. 197.

---

© Берецкий Игорь Сергеевич ( [ISberetskij@mai.ru](mailto:ISberetskij@mai.ru) ), Ирбитский Илья Сергеевич ( [ILSirbitskij@mai.ru](mailto:ILSirbitskij@mai.ru) ),  
 Егорова Евгения Кирилловна ( [egorovaek@mat.i.ru](mailto:egorovaek@mat.i.ru) ), Мокряков Алексей Викторович ( [MokryakovAlVik@gmail.com](mailto:MokryakovAlVik@gmail.com) ),  
 Чернова Татьяна Александровна ( [Chernovata@mai.ru](mailto:Chernovata@mai.ru) ).  
 Журнал «Современная наука: актуальные проблемы теории и практики»