**Санкт–Петербургский государственный университет**

***СКОРОХОДОВА Влада Валентиновна***

**Выпускная квалификационная работа**

# *Применение методов теории графов для решения прикладных задач*

Уровень образования: бакалавриат
Направление 01.03.02 «Прикладная математика и информатика»
Основная образовательная программа СВ.5005 «Прикладная математика, фундаментальная информатика и программирование»
Профиль «Прикладная математика, информатика и процессы управления»

Научный руководитель:
доцент, кафедра высшей математики,
д.ф. - м.н. Калинина Елизавета Александровна

Рецензент:
доцент, кафедра компьютерных технологий
и систем, к.ф. - м.н. Погожев Сергей Владимирович

Санкт-Петербург
2020 г.

# Содержание

# Введение

В современном развивающемся мире в абсолютно различных отраслях необходима работа с большими массивами данных. Например, в экономике, производственной или научной деятельности. Иногда эти данные необходимо упорядочить, и тогда разумно разделить их на кластеры, то есть группы, связанные между собой по какому-либо признаку.

Отдельный интерес представляет кластеризация матриц с неотрицательными элементами. Такие матрицы встречаются в различных разделах математики. Например, в теории вероятностей при исследовании цепей Маркова (стохастические матрицы) или в теории малых колебаний упругих систем (осцилляционные матрицы) [1]. Существует математический аппарат работы с неотрицательными матрицами, но не всегда известные алгоритмы возможно точно реализовать программно. Одним из таких алгоритмов является алгоритм проверки разложимости матрицы, то есть приведение матрицы к определенному виду с помощью перестановок строк и столбцов. Кроме того, известные алгоритмы, которые являются эффективными в случае матриц небольших размерностей, оказывается неприменимыми для больших матриц. Поэтому возникает необходимость разработки новых методов и алгоритмов, которые могут использоваться для решения практических задач, связанных с обработкой больших массивов данных, порождающих матрицы больших порядков.

Теория графов позволяет упростить и оптимизировать работу с неотрицательными матрицами, в том числе и вышеупомянутый алгоритм.

# Постановка задачи

В данной работе рассматривается задача построения нормальной формы матриц больших размеров с неотрицательными элементами, связанная с задачей разбиения графа на компоненты связаности и выявления несвязанных вершин в каждой компоненте и исследуется непосредственное ее практическое приложение для оптимизации рекламных кампаний в социальных сетях. Нормальная форма матрицы — это блочно-диагональная матрица, к которой можно привести исходную матрицу с помощью перестановок ее строк и столбцов.

Работа состоит из двух глав: в первой обсуждаются теоретические результаты, а во второй представлен реальный пример разделения группы с помощью предложенного алгоритма.

В первой главе представлены определения, теоремы и леммы из линейной алгебры и теории графов, решается задача улучшения алгоритма построения нормальной формы разложимой матрицы.

Во второй главе рассматривается следующая задача: разделить данное множество пользователей социальной сети на группы с целью эффективного рекламирования товара.

В приложении приводится программная реализация решения задачи из второй главы.

## Обзор литературы

Основная алгебраическая теория для данной работы была взята из [1]. Все определения теории графов были получены из книги [3] и электронного пособия Г.М. Хитрова и С.В. Погожева [5], которым автор выражает большую благодарность.

Для доказательства матричных теорем теория графов применялась в [2]. Эта работа прямо продолжает данное начинание.

Практическое применение результата, полученного в ходе работы, стало возможно благодаря теоретическим результатам Д.В. Савицкой, представленным в статье [7]. Ей принадлежит понятие нормальной формы графа, а также алгоритм выявления несвязанных вершин и построения нормальной формы.

# Глава 1. Теоретические результаты

## 1.1 Алгебраическая теория

Рассмотрим неотрицательную матрицу $A = (a_{ij})_{i,j=1}^n$, т.е. матрицу, все элементы которой неотрицательны: $a_{ij} \geq 0$.

Говорят, что квадратная неотрицательная матрица $A$ разложима, если с помощью перестановок ее рядов (то есть одновременных перестановок строк и столбцов) ее можно привести к виду

$$\tilde{A} = PAP^T = \begin{pmatrix} B & C \\ \mathbb{O} & D \end{pmatrix} \tag{1}$$

где $B$, $D$ — квадратные матрицы, а $P$ — матрица перестановок, то есть квадратная (0,1)-матрица, в каждой строке и в каждом столбце которой находится лишь одна единица. В противном случае неотрицательная матрица неразложима.

Неразложимые матрицы, в свою очередь, разделяются на два класса — примитивные и импримитивные. О том, каким образом происходит разделение, говорит следующая теорема (она также упоминается в литературе как спектральное свойство неразложимых неотрицательных матриц).

**Теорема 1.1** (Фробениус). *Неразложимая неотрицательная матрица $A$ размерностью $n \times n$ всегда имеет положительное характеристическое число $r$, которое является простым корнем характеристического полинома матрицы $A$. При этом модули всех других характеристических чисел не превосходят $r$ и $r$ соответствует собственный вектор $z$ с положительными координатами. Если при этом $A$ имеет $h$ характеристических чисел $\lambda_0 = r, \lambda_1, \ldots, \lambda_{n-1}$, по модулю равных $r$, то все эти числа различны между собой и являются корнями уравнения*

$$\lambda^h - r^h = 0$$

*Число $h$ называется индексом импримитивности матрицы $A$. При $h > 1$ перестановкой рядов можно привести матрицу $A$ к «циклическому»* *виду:*

$$\tilde{A} = PAP^T = \begin{pmatrix} 0 & A_{12} & 0 & \dots & 0 \\ 0 & 0 & A_{23} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & A_{h-1,h} \\ A_{h1} & 0 & 0 & \dots & 0 \end{pmatrix}$$

*Матрица, которую можно привести к циклическому виду, называется импримитивной. При* $h = 1$ *матрица* $A$ *называется примитивной. Также* $h$ *легко определить, если известны коэффициента характеристического уравнения* $f(\lambda) = \lambda^n + a_1\lambda^{n_1} + a_2\lambda^{n_2} + \dots + +a_t\lambda^{n_t} = 0 \quad (n > n_1 > \dots > n_t; a_1 \neq 0, a_2 \neq 0, \dots a_t \neq 0)$ *матрицы, а именно,* $h$ *равно НОД разностей* $n$ - $n_1, n_1 - n_2, \dots, n_{t-1} - n_t$.

Для примитивных матриц выполняется следующее важное свойство:

**Теорема 1.2.** *Матрица* $A \geq 0$ *является примитивной в том и только в том случае, когда некоторая ее степень положительна:*

$$A^p > 0 \quad (p \geq 1),$$

*то есть все элементы матрицы* $A^p$ *положительны.*

Таким образом, все положительные матрицы неразложимы и примитивны. Также заметим, что неразложимая матрица не может иметь нулевую строку или столбец, иначе ее всегда можно привести к виду (1). Если при приведении к виду (1) разложимой матрицы $A$ порядка $n > 1$ подматрица $C = 0$, то матрица $A$ называется сильно разложимой, в противном случае — слабо разложимой.

## 1.2 Переход от неотрицательных матриц к графам

Что нужно сделать для того, чтобы понять, к какому классу принадлежит неотрицательная матрица (и, соответственно, как с ней лучше работать в зависимости от поставленной задачи)?

Очевидно, нужно возводить матрицу в степень. Если найдется положительная степень матрицы, то исходная матрица будет неразложима и примитивна. Если же нет, и при этом сохраняется нулевой блок — матрица разложима. Неразложимую импримитивную матрицу можно возводить в степень бессчетное число раз.

Получается, что необходимо найти следующую оценку: до каких пор нужно возводить матрицу в степень, чтобы узнать, примитивна она или нет? В какой момент необходимо остановиться?

Хорн и Джонсон предлагают следующее решение.

Во-первых, проверку заданной неотрицательной матрицы на примитивность можно проводить, выясняя неразложимость данной матрицы и используя критерий Вилландта.

**Лемма 1.3.** *Неотрицательная квадратная матрица $A$ размерностью $n \times n$ неразложима тогда и только тогда, когда $(E + A)^{n-1} > 0$.*

**Лемма 1.4** (Критерий Вилландта)**.** *Неотрицательная неразложимая матрица $A$ размерностью $n \times n$ примитивна тогда и только тогда, когда $A^{n^2-2n+2} > 0$.*

Также существует следующий критерий примитивности, существенно упрощающий задачу:

**Лемма 1.5.** *Если все элементы главной диагонали неотрицательной и неразложимой матрицы $A$ размерностью $n \times n$ положительны, то*

$$A^{n-1} > 0.$$

Число матричных умножений сокращается, если последовательно возводить матрицу в квадрат: во 2 степень, в 4, в 8, в 16, в 32 и т.д. до тех пор, пока не будет получена степень, превышающая критическое значение для одного из заданных критериев.

Но при программной реализации данной задачи может возникнуть следующая проблема. Элементы матрицы $A$ могут быть сколь угодно большими, что при нахождении особенно больших степеней может вызвать переполнение памяти и, соответственно в лучшем случае неверный ответ. Также, если элементы, наоборот, очень близки к нулю, то при каждом новом умножении они будут становиться все меньше и меньше до тех пор, пока не превратятся в 0 (числа с плавающей точкой записываются в памяти компьютера таким образом, что при перемножении возникает сдвиг и, соответственно, погрешность).

Чтобы избежать подобных казусов, попробуем упростить метод решения с помощью индикаторных матриц и теории графов.

Пусть у нас есть квадратная неотрицательная матрица $R$ с элементами $r_{ij}$, которую необходимо проверить на примитивность. Сопоставим ей индикаторную матрицу $A$ с элементами $a_{ij}$, такую, что

$$a_{ij} = \begin{cases} 0, & r_{ij} = 0 \\ 1, & r_{ij} > 0 \end{cases}$$

Или, другими словами, каждый ненулевой элемент заменим на единицу. Тогда мы получим квадратную (0,1)-матрицу. Из теории графов известно, что любая квадратная (0,1)-матрица задает ориентированный граф, для которого эта матрица, в свою очередь, является матрицей смежности.

В данном случае графом матрицы $A$ назовем такой граф $G(A)$, для которого $A$ — матрица смежности.

Получается, что любой неотрицательной матрице $R$ можно сопоставить граф, построенный на соответствующей индикаторной матрице.

Для (0,1)-матриц известно следующее утверждение.

**Лемма 1.6.** *(0,1)-матрица $A = \{a_{ij}\}$ размерности $n \times n$, где $n > 1$, неразложима тогда и только тогда, когда любые индексы $i$ и $j$ связаны в матрице $A$.*

Здесь индекс $i$ называется связанным с индексом $j$ в (0,1)-матрице $A = \{a_{ij}\}$, если существует такая цепочка индексов $\{k_0, k_1, ..., k_l\}$, $k_0 = i$, $k_l = j$, что $a_{k_s k_{s+1}} = 1$ для всех $s \in \{0, 1, ..., l-1\}$. Как можно интерпретировать это утверждение? Если любые два индекса в матрице смежности связаны, то эта матрица задает связный ориентированный граф. Для проверки графа на связность существует операция транзитивного замыкания. Таким образом, чтобы проверить неотрицательную квадратную матрицу на неразложимость, необходимо построить для нее индикаторную матрицу, а затем работать с той, как с матрицей графа.

## 1.3 Транзитивное замыкание

Как известно, граф можно представить как бинарное отношение между множествами вершин графа $V(G)$ и его ребер $R(G)$. В теории множеств существует операция транзитивного замыкания, которую можно описать так: транзитивное замыкание бинарного отношения $R$ на множестве $X$ есть наименьшее транзитивное отношение на множестве $X$, включающее $R$.

Другими словами, транзитивное замыкание дополняет отношение до транзитивности, как бы «выявляет пропущенные шаги». Но что получится, если применить операцию транзитивного замыкания к графу?

Прежде, чем говорить о результатах, которые возможно получить, определим понятие связности графа. Ориентированный граф называется связным, если между любыми двумя его вершинами существует как минимум один путь. Под путем подразумевается последовательность ребер, соединяющих смежные вершины. Граф называется сильно связным, если между любыми двумя его вершинами существует ориентированный путь.

Также для ориентированных графов существует классификация по типам связности. Помимо сильной связности существует односторонняя связность: если для двух любых вершин по крайней мере одна достижима из другой, а также слабая связность: таковая присутствует, если при отбрасывании направлений у ребер получается связный обыкновенный граф. Примеры того, как могут выглядеть различно связные графы, можно увидеть на рисунке ниже.


**Рис. 1:** Сильно связный, односторонне связный и слабо связный орграфы

Построение транзитивного замыкания для матрицы смежности графа по сути — это достраивание ребер, что и есть то самое «выявление пропущенных шагов». Тогда, если орграф сильно связный, то получится достроить смежные ребра для каждых двух вершин. Если же в графе существуют изолированные точки, то между ними нет никаких «пропущенных шагов», то есть ребер, и граф оказывается несвязным.

Что это будет означать, если мы будем работать с графами в матричном виде? Это будет означать то, что если граф является связным, то матрица его транзитивного замыкания будет состоять из одних единиц. Действительно, матрица, состоящая из одних единиц, будет определять граф, все вершины которого связаны между собой. Если же при построении транзитивного замыкания в получившейся матрице остаются нули, то тогда граф не является сильно связным и относится к одному из типов, представленных выше (односторонне, слабо связным или несвязным).

При построении транзитивного замыкания можно выяснить вид графа, если с помощью матрицы перестановок $P$ матрицу смежности транзитивного графа $T$ привести к каноническому виду $\tilde{T} = PTP^T$. Классификация по типам будет выглядеть следующий образом:

- **Матрица транзитивного замыкания состоит из одних единиц.** Тогда граф $G(A)$ сильно связный.

$$\tilde{T} = \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix}$$

- **Матрица транзитивного замыкания содержит нули, но не содержит симметрично расположенные относительно главной диагонали нули.** Тогда граф $G(A)$ односторонне связный.

- **Матрица транзитивного замыкания не является сильно разложимой матрицей и содержит симметрично относительно главной диагонали расположенные нули.** Тогда граф $G(A)$ слабо связный.

- **Матрица не приводится ни к одному из выше рассмотренных случаев.** Тогда граф $G(A)$ несвязный.

Под нормальной формой (0,1)-матрицы будем понимать такую форму матрицы, транзитивное замыкание которой совпадает с одной из нормальных форм транзитивной матрицы.

Существует два основных подхода поиска транзитивного замыкания. Первый из них основан на возведении индикаторной матрицы $A$ в степень. Но, как уже упоминалось в самом начале, реализация возведения матрицы в степень сопряжена с рисками погрешности вычислений.

Другим алгоритмом для поиска транизитивного замыкания служит алгоритм Флойда-Уоршелла, разработанный в 1962 году Робертом Флойдом и Стивеном Уоршеллом. Этот алгоритм служит для нахождения кратчайших расстояний в графе, но его возможно адаптировать и для нахождения транзитивного замыкания, если использовать булеву арифметику. Рассмотрим индикаторную (0,1)-матрицу не как (0,1)-матрицу в поле вещественных чисел, а перейдем к логической арифметике, в которой 1 означает истину, а 0 — ложь. Для работы с элементами матрицы будем использовать операции конъюнкции ($\wedge$ — логическое умножение) и дизъюнкции ($\vee$ — логическое сложение).

Будем полагать для матрицы транзитивного замыкания $T$ ее элемент $t_{ij}$ равным 1, если в графе $G(A)$ вершины $i$ и $j$ связаны между собой, и равным 0, если они не связаны. Тогда для построения «возможных» ребер используется следующее логическое соотношение:

$$t_{ij} = t_{ij} \vee (t_{ik} \wedge t_{kj})$$

## 1.4 Построение нормальной формы

Рассмотрим задачу построения нормальной формы матрицы с неотрицательными элементами. Для этого будем использовать алгоритм, предложенный Д.В. Савицкой из [6]. Изменение алгоритма заключается в том, что неотрицательной матрице $R$ сопоставляется индикаторная (0,1)-матрица $A$, а транзитивное замыкание находится с помощью алгоритма Флойда-Уоршелла.

1. Для данной неотрицательной матрицы $R$ построить матрицу $T$ транзитивного замыкания графа $G(A)$, связанного с $A$.

2. Если $T$ состоит только из единиц, тогда $A$ неразложима. Иначе, перейти к шагу 3.

3. Вычислить вектор $S$, состоящий из сумм строк $T$, затем построить матрицу перестановок $P$, такую, что $PS = \tilde{S}$, где $\tilde{S}$ — вектор, в котором элементы отсортированы в порядке невозрастания. Вектор $\tilde{S}$ содержит $k$ групп равных значений с $n_j$ $(j = 1, 2, \ldots, k)$ значениями в каждой $j$-ой группе, $\sum_{j=1}^{k} n_j = n = \dim(A)$. Отсюда получить матрицу
$$\mathcal{T} = PTP^T = \begin{bmatrix} T_1 & 0 & \ldots & 0 \\ 0 & T_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & T_k \end{bmatrix}$$
где $T_j$ — квадратная матрица размерности $n_j$. Каждая матрица $T_j$ будет указывать на соответствующую компоненту связности графа. Если граф связный, то в матрице $\mathcal{T}$ будет только один блок.

4. Рассмотреть матрицы $T_j$, найти в них нули, симметричные относительно главной диагонали. Если $T_j$ не имеет таких нулей, тогда соответствующая компонента матрицы $A$ дает односторонне связный граф $G(A_j)$. В противном случае граф $G(A_j)$ слабо связный, и мы получаем те пары вершин, которые не связаны друг с другом. Если же $T_j$ состоит полностью из единиц, то соответствующий граф — сильно связный.

5. Найти матрицы
$$\tilde{A} = PRP^T = \begin{bmatrix} A_1 & 0 & \ldots & 0 \\ 0 & A_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & A_k \end{bmatrix}, \tilde{R} = PRP^T = \begin{bmatrix} R_1 & 0 & \ldots & 0 \\ 0 & R_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & R_k \end{bmatrix}.$$

В качестве результата должна получиться матрица в виде (1), представленная в начале главы. Параллельно с нахождением канонической формы представляется возможным выявить вершины, не связанные друг с другом (из пункта 4 алгоритма).

Матрица перестановок будет верной как для $A$, так и для $R$, поскольку $R$ по своей сути представляет матрицу смежности графа с весами.

**Пример.** Рассмотрим в качестве примера следующую матрицу:

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

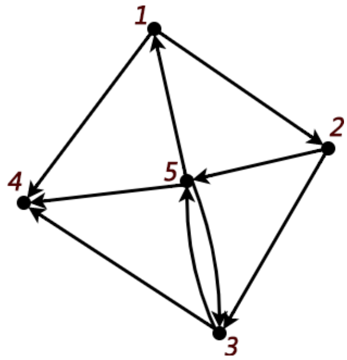


**Рис. 2:** Граф $G(A)$

Применим вышеописанный алгоритм к матрице $A$. Поскольку это уже (0,1)-матрица, индикаторную матрицу строить не нужно.

1. Матрица транзитивного замыкания графа $G(A)$ будет выглядеть следующим образом:
$$T = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

2. Матрица $A$ разложима.

3. 
$$S = \begin{pmatrix} 5 \\ 5 \\ 5 \\ 0 \\ 5 \end{pmatrix}, \tilde{S} = \begin{pmatrix} 5 \\ 5 \\ 5 \\ 5 \\ 0 \end{pmatrix}, P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \mathcal{T} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

   Матрица $\mathcal{T}$ состоит из одного блока, поэтому можно видно, что граф — связный.

4. Поскольку в матрице $\mathcal{T}$ не существует нулей, симметричных относительно главной диагонали, граф $G(A)$ — односторонне связный (что

можно видеть из рисунка).

5.

$$\tilde{A} = PAP^T = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

# Глава 2. Практическое приложение

В качестве практического приложения алгоритма рассмотрим задачу о рекламировании в социальных сетях. Пусть у нас есть группа профилей, связанных друг с другом через отношение «друзья». Задача состоит в следующем: как наиболее оптимально распределить таргетированную рекламу/товар, показав ее наименьшему числу пользователей, чтобы впоследствии товар купило наибольшее число людей? Предполагается, что пользователь, увидевший рекламу/получивший товар, расскажет о нем другим. Чтобы решить эту задачу, воспользуемся алгоритмом, представленным в предыдущей главе. Таким образом, мы выясним, какие пользователи не связаны друг с другом.

Для данной работы была взята группа профилей в социальной сети ВКонтакте. Сначала были проанализированы 127 друзей отдельно взятого пользователей. Связи между ними были представлены в виде графа с матрицей смежности $A$.

Следуя алгоритму, описанному в предыдущей главе, была построена матрица транзитивного замыкания $T$. Поскольку она содержит нули, то матрица $A$ — разложима, что позже подтвердилось при проверке с помощью шага 5 вышеописанного алгоритма.

Вектор $\tilde{S}$ получился следующим:

$$\tilde{S} = \begin{bmatrix} 105 & \dots & 105 & 2 & 2 & 0 & \dots & 0 \end{bmatrix}^T$$

где количество элементов в каждой группе 105 для $n_1 = 105$, 2 для $n_2 = 2$ и 20 для $n_3 = 0$.

В качестве следующего шага была найдена матрица перестановок $P$. Матрица перестановок находится следующим образом: рассматривается единичная матрица, затем с помощью двойного перебора на каждом шаге проверяется, является ли строка $P_i$ такой, что ее умножение на $S$ дает соответствующий элемент $\tilde{S}_i$. Если да, то строка остается в том же положении, если нет, то она переставляется с другой до тех пор, пока не найдется нужная.

Затем была составлена матрица $\mathcal{T}$. Она содержит всего лишь один блок $T_1$, что означает, что в графе $G(A)$ всего одна компонента связности. $\mathcal{T}$ содержит 2540 нулей, симметричных относительно главной диагонали, что означает, что граф $G(A)$ слабо связный. Поскольку привести все пары несвязанных вершин было бы трудно из-за их количества, приведу получившиеся пары только для одной точки. Вершина 1, например, не связана с вершинами 5, 9, 10, 15, 17, 24, 35, 42, 44, 48, 68, 73, 76, 79, 86, 104, 114, 119, 120, 123, 124, 125.

Все вычисления были выполнены в среде Matlab R2019b, данные были получены с помощью программы на языке Python. Токен доступа к ВКонтакте был получен с помощью кода из работы [9].

Вся программная реализация представлена в приложении.

Далее, после выявления несвязанных друг с другом пользователей, целесообразно определить долю пользователей, которым нужно отправить товары в качестве рекламы. Соответствующий алгоритм приведен в работе [8]. После нахождения доли пользователей можно воспользоваться одним из алгоритмов для нахождения лидера в каждой группе, чтобы реклама получилась еще более эффективной. Один из таких алгоритмов представлен в работе [6].

## Заключение

В работе была рассмотрена задача построения канонической матрицы для неотрицательной матрицы большого порядка. Уже существующий алгоритм решения задачи был улучшен: ранее при проверке матрицы на разложимость могло возникнуть переполнение в случае больших значений в оригинальной матрице либо, наоборот, в случае значений, близких к нулю. В новом алгоритме этой проблемы нет, поскольку проверяется индикаторная матрица, и используются только матрицы, состоящие из нулей и единиц.

Недостаток метода заключается в том, что его можно применять только к квадратным матрицам.

Помимо этого была решена задача выявления несвязанных вершин в графе. Применение результата алгоритма было рассмотрено во второй главе для решения оптимизационной задачи рекламы в социальных сетях.

# Список литературы


[1] Гантмахер Ф.Р. Теория матриц. Изд-во «Наука», М., 1967. 578 с.

[2] Хорн Р., Джонсон Ч. Матричный анализ. Изд-во «Мир», М., 1989. 656 с.

[3] Харари Ф. Теория графов. Изд-во «Мир», М., 1973. 302 с.

[4] Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Ривест, Клиффорд Штайн. Алгоритмы: построение и анализ. — 2-е изд. — М., Изд-во «Вильямс», 2006. — С. 1296

[5] Погожев С.В., Хитров Г.М. Теория графов. Учебное пособие

[6] Gubanov, D.A., Novikov, D.A., Chkhartishvili, A.G.: Informational influence and informational control models in social networks. Autom. Remote Control 72, 1557–1567 (2011)

[7] Савицкая Диана Владимировна Нормальная форма (0,1)-матрицы и алгоритмы ее построения // Вестник СПбГУ. Серия 10. Прикладная математика. Информатика. Процессы управления. 2008

[8] Zhao, B., Li, Y.K., Lui, J.C.S., Chiu, D.-M.: Mathematical Modeling of Advertisement and Influence Spread in Social Networks. In NetEcon: Workshop on the Economics of Networks, Systems and Computation (2009)

[9] Research Repository Saint-Petersburg State University, https://dspace.spbu.ru/bitstream/11701/4373/1/Diplom3.pdf

# Приложение

**friends.py**

```python
import requests
import time
import numpy as np
import scipy.io
// токен необходимо вставить ниже
token = '9f4809c3ee151c391781d9408ba915ea09fa5d9'
req = 'https://api.vk.com/method/friends.get?order=name&
fields=nickname,photo_200_orig,education&v=5.0&access_token=' + token

resp = requests.get(req)
friends = list(filter(lambda x: not x.get('deactivated'),
resp.json()['response']['items']))

matrix = []
for f in friends:
    print(f)
    req = 'https://api.vk.com/method/friends.getMutual?target_uid=' + str(f['id']) + '&v=5.0&access_token=' + token
    resp = requests.get(req)
    mutual = resp.json()['response']
    row = []
    for _f in friends:
        if _f['id'] in mutual:
            row += [1]
        else:
            row += [0]
    matrix.append(row)
    time.sleep(1)
```

```
matrix = np.array(matrix)
print(matrix)
scipy.io.savemat('output.mat', {'arr': matrix})
```

**algorithm.m**

```
load output.mat
%output.mat - файл, содержащий матрицу смежности arr
%алгоритм Флойда-Уоршелла
t = logical(arr);
for k =1:size(arr, 1)
    for i =1:size(arr, 1)
        for j = 1:size(arr, 1)
            t(i, j) = t(i, j) || (t(i, k) && t(k, j));
        end
    end
end
n = isirred(t);
if n==0:
    s = zeros(size(arr, 1), 1);
    sum = 0;
    for i = 1:size(t, 1)
        sum = 0;
       for j = 1:size(t, 1)
           sum = sum + double(t(i, j));
       end
       s(i) = sum;
    end
    stilda = sort(s, 'descend');
    P = permu(s, stilda);
    D = zeros(size(arr));
    for i = 1:size(D, 1)
       for j = 1:size(D, 1)
```

```
            D(i, j) = i*1000+j;
        end
    end
    F = P*double(t)*P';
    FD = P*D*P';
%переменная vertices содержит маркеры,
%которые указывают на несвязанные вершины
    vertices = [];
    for i = 1:size(F, 1)
        for j = i+1:size(F,1)
            if F(i, j)==F(j, i) && F(i, j) == 0
               vertices = [vertices FD(i, j)];
            end
        end
    end
else
    disp("The matrix A is irreducible");
end
```

**isirred.m**

```
%эта функция получает на вход матрицу T и выдает логический ответ
%true/false на вопрос
%"Является ли матрица неразложимой?"
function ans = isirred(matrix)
f = 0;
for i = 1:size(matrix, 1)
   for j = 1:size(matrix, 1)
       if matrix(i, j)==false
           f = f + 1;
       end
   end
end
if f==0
```

```
    ans = true;
else
    ans = false;
end
```

**permu.m**

```
%эта функция берет векторы s и stilda
%и выдает матрицу перестановок P, такую, что
%s*P = stilda
function P = permu(s, stilda)
P = eye(length(s));
for i = 1: size(s)
    for k = i+1:size(s)
        p = P(i, :).*s;
        sum = p(1);
        if sum ~= stilda(i)
            P([i,k],:) = P([k,i],:);
        else
            break
        end
    end
end
end
```