

В.Н. КУДЕЛЯ

**РЕШЕНИЕ ЗАДАЧ ПЕРЕБОРА ПУТЕЙ В СЛОЖНЫХ ГРАФАХ***Куделя В.Н. Решение задач перебора путей в сложных графах.*

**Аннотация.** Моделирование различных систем связано с перебором значений параметров элементов структуры и учетом всех характеристик функционирования и взаимодействия компонентов для нахождения определенного набора решений, определяющих конфигурацию системы. Такие задачи относятся к задачам переборного типа и подразумевают, что некоторое количество очередных решений из этого набора получается из предыдущего решения в определенном порядке. Известно, что достаточно большое количество задач переборного типа решается только методами полного перебора и других методов для их точного решения пока не существует. В статье представлен новый метод перебора путей в графе – метод трансформации узлов-графов. По предварительной оценке, предложенный метод, в отличие от существующих, позволяет значительно быстрее осуществлять поиск всех простых путей в ориентированном графе произвольной структуры. В известных методах перебора в графе (Breadth First Search и Depth First Search) объектом перебора является путь. Всё количество таких путей в графе определяет размер пространства перебора. Основная идея метода трансформации узлов-графов заключается в значительном уменьшении размера пространства перебора за счет укрупнения объектов перебора. Укрупнение объектов перебора осуществляется кластеризацией путей в комбинаторные объекты, объединяющие по определенному регламенту некоторое множество путей одинаковой длины. Такие комбинаторные объекты названы узлами-графами. Узел-граф относится к центрально-периферическим комбинаторным объектам и для перебора всех путей в графе разработаны специфические операции преобразования узлов-графов, которые позволяют найти следующие пути на основе предыдущих. Метод может использоваться как базовый инструментальный для уменьшения размерности пространства поиска решений NP-полных задач, сохраняя универсальность и точность перебора.

**Ключевые слова:** граф, путь, гамильтонов путь, перебор, комбинаторный взрыв, NP-полные задачи, кластеризация, узел-граф.

**1. Введение.** С бурным развитием информационных технологий теория графов резко расширило сферу своего применения в различных областях знаний. Графы получили исключительную популярность при моделировании таких систем, как распределенные бухгалтерские книги типа направленных ациклических графов [1], социальные сети [2], транспортные системы [3, 4], системы связи [5], системы комбинаторных аукционов [6], финансовые системы [7] и т.д.

Моделирование всех компонентов системы, таких как структура, регламенты, алгоритмы функционирования и других, обычно сводится к решению комплекса задач теории графов, которые можно условно объединить в несколько групп:

– задачи анализа связности: оценка наличия связи между какими-либо элементами графа;

– задачи упорядочения вершин или задачи кластеризации: выделение неких скоплений в графе. Например, независимые множества вершин или множества вершин с одинаковыми характеристиками и т.д.;

– задачи размещения центров: выделение каких-либо элементов графа с уникальными свойствами, например, минимаксное множество центров, множество центров с минимальной суммой и т.д.;

– задачи о маршрутах: путь, гамильтонов путь, независимые пути и т.д.;

– потоковые задачи: задача о максимальном потоке, поток в сети с ограничениями, распараллеливание потока задач, распределение потоков данных и т.д.

Однако известно [8], что наибольшее количество практически важных задач теории графов входят в список NP-полных задач. Кроме того, любая NP-полная задача может быть решена методом полного перебора [9 – 11], например, задача перечисления путей. Количество путей длиной от  $p = 1$  до  $p = v - 1$  в полном ориентированном графе составляет  $ev!$  ( $e = 2,71828\dots$  – число Эйлера,  $v = |V|$  – мощность множества вершин  $V$  графа  $G[V, E]$ ). Из-за такого количества результатов (размера пространства перебора) задача перечисления путей относится к задачам с комбинаторным взрывом [12], что и определяет высокий уровень сложности большинства NP-полных задач.

Наиболее известными являются два метода перебора путей в графе – поиск в ширину (Breadth First Search, BFS) и поиск в глубину (Depth First Search, DFS). А метод возведения в степень матрицы смежности такого распространения не получил [13]. Методы BFS и DFS исследованы в достаточном объеме и их использование при решении многих задач переборного типа приводит к комбинаторному взрыву. Комбинаторный взрыв происходит, когда создается громадное количество возможных комбинаций за счет увеличения количества объектов перебора. Для обхода проблемы комбинаторного взрыва разрабатывают специальные методы решения. Чаще всего применяют эвристические алгоритмы, которые основываются на предположениях о наиболее перспективных ветвях процесса перебора, остальные ветви исключаются. Исходя из необходимости преодоления комбинаторного взрыва, автор разработал и исследовал несколько методов перебора путей в графе и способов кластеризации результатов перебора:

1. Разработаны следующие методы перебора:

– метод транспонирования [14], позволяющий решать задачу поиска всех путей в графе, так же как и методы BFS и DFS, перебирая пути длины  $p$  для поиска путей длины  $p + 1$ ;

– инверсный метод [15], дающий возможность перебирать пути длиной  $p$  для поиска путей длиной  $p + k$ , где  $k = \{1, 2, 3, \dots\}$  – размер прироста (шага) длины пути. Пространство перебора сократилось и при размере шага  $k = 2$  составило  $\frac{8\sqrt{3}}{9}v!$ , где  $\frac{8\sqrt{3}}{9}$  – квадратная ледяная постоянная *Либа*.

2. Исследование разработанных методов позволило подтвердить некоторые причины комбинаторного взрыва и сформулировать новые положения по ускорению алгоритмов перебора:

– вычислительная сложность решения задач поиска всех путей в графе обусловлена тем, что чем больше длина  $p$  отыскиваемых путей, тем больше их возможное количество;

– количество искомым путей длиной  $p + 1$ , находится в зависимости вида  $O_{p+1} = O_p(v - p - 1)$  от количества путей длиной  $p$ ;

– сложность решения NP-полных задач может быть снижена, например при использовании инверсного метода, для решения задачи поиска всех гамильтоновых путей в графе. При этом точность полученных результатов сохраняется;

– один из основных способов значительного снижения эффекта комбинаторного взрыва – это сокращение пространства перебора, но не с помощью эвристик [16 – 20], дающих вероятные результаты, а за счет объединения путей в группы (кластеры), то есть объединить в более крупные объекты перебора [14, 15];

– дополнительно сократить пространство перебора и, как следствие, ускорить алгоритм перебора можно при увеличении, где это допустимо, размера шага  $k = \{2, 4, 6, \dots\}$ .

В данной статье представляется новый метод перебора путей в графе – метод трансформации узлов-графов. Для демонстрации метода трансформации узлов-графов автор выбрал задачу перечисления гамильтоновых путей в графе [18]. Постановка задачи: какова мощность множества гамильтоновых путей в графе  $G[V, E]$ .

**2. Общие положения.** Автор использовал следующие термины.

*Путь* – последовательность ребер  $e_1, e_2, \dots, e_k$ , в которой конец одного ребра является началом следующего ребра. Начало первого ребра называется началом пути, конец последнего ребра – концом пути.

*Простой путь* – путь, который проходит через каждую вершину не более одного раза.

*Длина пути*  $p$  – число ребер (дуг), которые его образуют.

*Гамильтонов путь* – это простой путь, проходящий через все вершины графа.

*Окрестность вершины*  $v_k$  в графе – подграф графа  $G[V, E]$ , состоящий из всех вершин, сопряженных  $v_k$ , и всех ребер, соединяющих две такие вершины.

*Узел-граф* – подграф графа  $G[V, E]$ , состоящий из образующей подмножество  $U$  последовательности вершин  $u_1, u_2, \dots, u_i$  ( $i = \{1, 2, 3, \dots\}$ ,  $i \leq v - 2$ ), из подмножества  $S$  вершин, не входящих в  $U$  и имеющих исходящую дугу в  $u_1$ , и из подмножества  $T$  вершин, не входящих в  $U$  и имеющих исходящую дугу из  $u_i$ , а также всех дуг, соединяющих такие вершины с первой  $u_1$  и последней  $u_i$  из подмножества  $U$ .

*Схема кластеризации* – схема группирования комбинаторных объектов, в нашем случае путей.

В дальнейшем, если не оговаривается иное, под графом понимается ориентированный граф без петель. В графе ищутся простые пути, далее – путь. Для обозначения вершины будем использовать одну цифру, а цифры через запятую – это дуга, соединяющая смежные вершины. Точка с запятой разделяет числа, обозначающие вершины, не имеющие признака смежности в подмножестве.

На изображениях графа, для того чтобы не загромождать рисунок, ненаправленными ребрами будут обозначаться две противоположно направленные дуги.

### **3. Метод трансформации узлов-графов**

**3.1. Представление графа.** Новое представление графа будет далее рассмотрено на основе примера графа, изображенного на рисунке 1.

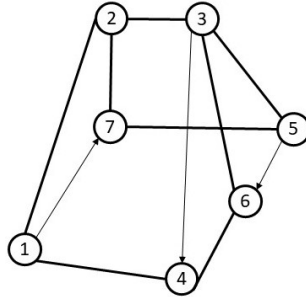


Рис. 1. Граф  $G[V, E]$

Выделим в графе  $G[V, E]$  одну из окрестностей, например для вершины 5, смотри рисунке 2.

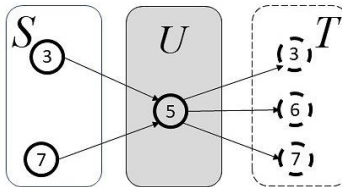


Рис. 2. Узел-граф

Вершины из окрестности (сплошной контур), имеющие исходящую дугу, направленную к вершине с серой заливкой, размещаем слева от вершины с серой заливкой. Это подмножество вершин назовем истоком  $S$  вершины с серой заливкой. Вершины из окрестности (штрих-контур), имеющие входящую дугу, направленную из вершины с серой заливкой, размещаем справа от вершины с серой заливкой. Подмножество вершин, имеющих штрих-контур, будем называть стоком  $T$  вершины с серой заливкой.

Будем считать, что путь от какой-либо вершины в себя существует всегда – это путь длиной  $p = 0$ . Исходя из этого, примем, что область с серой заливкой тоже является подмножеством – телом  $U$  узла-графа. Кластер подмножеств  $S$ ,  $U$ ,  $T$  и является узлом-графом. В отличие от окрестности вершины, узел-граф связан с местом сосредоточения путей, таких как перекресток, железнодорожный узел, центр города, где пересекаются пути. То есть узел-граф – подграф

графа  $G[V, E]$ , состоящий из образующей подмножество  $U$  последовательности вершин  $u_1, u_2, \dots, u_i$  ( $i = \{1, 2, 3, \dots\}$ ,  $i \leq v - 2$ ), из подмножества  $S$  вершин, не входящих в  $U$  и имеющих исходящую дугу в  $u_1$ , и из подмножества  $T$  вершин, не входящих в  $U$  и имеющих исходящую дугу из  $u_i$ , а также всех дуг, соединяющих такие вершины с первой  $u_1$  и последней  $u_i$  из подмножества  $U$ .

Символьное представление узла-графа имеет вид  ${}^{(q)}U_{[S]}^{(w)}_{[T]}$ .  $q$  и  $w$  будут определены далее. Схема кластеризации путей в узел-граф представлена в таблице 1.

Таблица 1. Схема кластеризации путей в узел-граф

№ п/п	Отношения в подмножествах и между подмножествами
1.	Тело узла-графа $U$ – подмножество последовательности смежных вершин. $U \subset V$ , $u =  U $ – мощность множества $U$ , $u = \{1, 2, 3, \dots\}$ , $u \leq v - 2$
2.	Исток узла-графа $S$ – подмножество несвязанных вершин. $S \subset V$ , $s =  S $ – мощность множества $S$ , $s = \{0, 1, 2, 3, \dots\}$ , $s \leq (v - 1)$
3.	Исток узла-графа размещается слева от тела. Вершина из истока $S$ имеет только исходящую дугу, направленную к первой вершине из тела $U$ , $S \cap U = \emptyset$
4.	Сток узла-графа $T$ – подмножество несвязанных вершин. $T \subset V$ , $t =  T $ – мощность множества $T$ , $t = \{0, 1, 2, 3, \dots\}$ , $t \leq (v - 1)$
5.	Сток размещается справа от тела. Вершина из стока $T$ имеет только входящую дугу, направленную от последней вершины из тела $U$ , $T \cap U = \emptyset$

Для удобства демонстрации операций с узлами-графами, элементы подмножеств  $S$  и  $T$  могут располагаться либо горизонтально через точку с запятой, либо – вертикально без разделительных знаков.

Параметры  $q$  и  $w$  характеризуют узел-граф. Данные параметры являются дополнительными, их наличие и количество необязательно. Это могут быть:

- внутренние параметры узла-графа: количество элементов, вес, длина, стоимость и другие;
- параметры моделируемого объекта: параметры взаимодействия со средой, в которой функционирует объект; технические; химические; биоинформационные; генетические и т.д. характеристики объекта.

Совокупность всех узлов-графов, у которых мощность множества  $u = 1$ , полностью описывает граф  $G[V, E]$ , то есть является способом представления графа, эквивалентным всем существующим способам, так как на основе любого представления можно получить все остальные. Каждый узел-граф, у которого мощность множества  $u = 1$ , объединяет все направленные пути длиной  $p = 2$ , проходящие через тело  $U$ . Все пути в узле-графе направлены от истока  $S$  к стоку  $T$ .

Тогда граф  $G[V, E]$  (рисунок 1) задается следующей совокупностью узлов-графов:

$$\begin{matrix} \binom{(4)}{1}^{(1)} & \binom{(6)}{2}^{(2)} & \binom{(9)}{3}^{(3)} & \binom{(4)}{4}^{(4)} & \binom{(4)}{5}^{(5)} & \binom{(4)}{6}^{(6)} & \binom{(4)}{7}^{(7)} \\ [2;4] & [1;3;7] & [2;5;6] & [1;3;6] & [3;7] & [3;4;5] & [1;2;5] \end{matrix}$$

Для вершин из подмножеств  $S$  и  $T$  признак смежности не указывается, даже если они являются смежными в графе.

Здесь и далее  $q$  – число путей, проходящих через тело  $U$ , а  $w = \{1, 2, 3, \dots\}$  – номер узла-графа в совокупности.

Количество путей длины  $p$  в графе можно рассчитать по формуле  $O_p = \frac{v!}{(v-p-1)!}$ . Тогда в полном ориентированном графе,

состоящем из семи вершин, имеется 210 путей (объектов перебора) длиной  $p = 2$ . При этом каждый путь содержит 3 вершины (элемента). То есть все пути включают 630 элементов. Тот же граф, представленный узлами-графами, включает 7 объектов перебора. Каждый узел-граф может содержать максимум 13 элементов. Тогда представление графа совокупностью узлов-графов содержит 91 элемент, что почти в 7 раз меньше. С увеличением числа вершин в графе степень отличия возрастает.

Кластеризация элементов (вершин, дуг) множеств  $V$  и  $E$  в группы подмножеств  $S$ ,  $U$  и  $T$  позволяет представить граф в более компактной форме – в виде совокупности узлов-графов. Очевидно, что количество объектов перебора длиной  $p$  для поиска объектов длиной  $p+k$  уменьшилось. Таким образом, пространство перебора за счет укрупнения объектов перебора сократилось.

Кластеризация может производиться и по другим основаниям, а узел-граф может создаваться и для других объектов перебора. Внутри подмножеств  $S$ ,  $U$  и  $T$  также могут формироваться группы объектов.

**3.2. Операции трансформации узла-графа.** Здесь представлены операции трансформации узла-графа для поиска всех путей в совокупности узлов-графов. Операции трансформации узла-графа позволяют найти пути длиной  $p+k$  на основе путей длиной  $p$ , то есть осуществить исчерпывающий поиск путей в графе.

При описании и демонстрации операций трансформации узла-графа подразумевается, что параметр  $w$ , номер узла-графа, относится ко всем составляющим узел-графа. Операции трансформации узла-графа выполняются для комбинаторных объектов, таких как вершина, дуга, узел-граф, находящихся в отношениях смежности. Узлы-графы  ${}_{[S]}^{(q)}U_{[T]}^{(1)}$  и  ${}_{[S]}^{(q)}U_{[T]}^{(2)}$  являются смежными, если вершина  $i \in T^{(1)}$  и  $i \in S^{(2)}$  или  $i \in T^{(2)}$  и  $i \in S^{(1)}$ .

Узел-граф относится к центрально-периферическим комбинаторным объектам, и процесс поиска путей в графе ассоциируется с последовательным поглощением центром (телом)  $U$  периферийных элементов (смежных узлов-графов, вершин из  $S$  и  $T$ ).

Любой узел-граф ( $u \geq 1$ ) разрешено использовать в операциях неоднократно, например, для увеличения шага  $k$ . Для перебора в графе каждый узел-граф выполняет функции то центра, то периферии. Любая операция трансформации может выполняться совершенно независимо для различных узлов-графов, то есть обеспечивается параллельный процесс получения результатов перебора.

Узел-граф, имеющий  $u > 1$ , существует при условии:

- если  $S^{(w)} \neq \emptyset \wedge T^{(w)} \neq \emptyset$ , путь обязательно где-то начинается и куда-то приводит;
- если  $s = t = 1$  и  $S^{(w)} \neq T^{(w)}$ , циклы исключаются.

**3.2.1. Добавление вершин.** Регламент добавления вершин в узел-граф представлен в таблице 2.



Таблица 2. Регламент добавления вершин в узел-граф

№ п/п	Регламент	Операции
1.	$d = \max w + 1$	+ – сложения
2.	Если $i \in S^{(w)}$ , $j \in T^{(w)}$ и $i \neq j$ , то $U^{(w=d)} = i + U^{(w)} + j$	+ – добавления вершины
3.	$S^{(w=d)} = S^{(w=i)} \setminus (S^{(w=i)} \cap U^{(w=d)})$	$\setminus$ – разность множеств
4.	$T^{(w=d)} = T^{(w=j)} \setminus (T^{(w=j)} \cap U^{(w=d)})$	$\cap$ – пересечения множеств
Проверка условий существования		
5.	узла-графа $\begin{bmatrix} j^{(w=d)} \\ S^{(w=d)} \end{bmatrix} U^{(w=d)} \begin{bmatrix} T^{(w=d)} \end{bmatrix}$	удаление узла-графа, если условие не выполняется
6.	Пересчет значения параметра $q$ узла-графа	

Существуют следующие ограничения на операцию добавления вершин:

- мощность множества вершин графа  $v$  не менее 5;
- операция выполняется для узлов-графов, у которых  $S^{(w)} \neq \emptyset$  и  $T^{(w)} \neq \emptyset$ .

Очевидно, что множество путей длиной  $p + 2$  получается из множества путей длиной  $p$ , то есть размер шага  $k = 2$ .

Добавление одной вершины с одной из сторон является частным случаем добавления вершин в узел-граф.

Алгоритм добавления вершин проиллюстрируем на узле-графе № 3 для графа, представленного на рисунке 1. В ячейки  $f_{i,j}$  таблицы  $F$  размером  $v \times v$ , у которых  $i \in S^{(3)}$ ,  $j \in T^{(3)}$  и  $i \neq j$ ,

вписывается узел-граф  $\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$ .

		Номера вершин стока $j$						
		1	2	3	4	5	6	7
Номера вершин истока $i$	1	0	0	0	0	0	0	0
	2	0	0	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0
	3	0	0	0	0	0	0	0
	4	0		0	0			0
	5	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0
	6	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	$\begin{bmatrix} q^{(3)} \\ 2,5;6^{(3)} \end{bmatrix} 3^{(3)} \begin{bmatrix} 2,4;5;6^{(3)} \end{bmatrix}$	0	0
	7	0	0	0	0	0	0	0

Далее выполняем регламент добавления вершин для узла-графа № 3. Например, для ячейки  $f_{2,4}$ , определим:

1.  $d = \max w + 1 = 7 + 1 = 8$ ;
2.  $U^{(8)} = i + U^{(3)} + j = 2 + 3 + 4 = 2, 3, 4$ ;
3.  $S^{(8)} = S^{(2)} \setminus (S^{(2)} \cap U^{(8)}) = 1; 3; 7 \setminus (1; 3; 7 \cap 2; 3; 4) = 1; 3; 7 \setminus 3 = 1; 7$ ;
4.  $T^{(8)} = T^{(4)} \setminus (T^{(4)} \cap U^{(8)}) = 1; 6 \setminus (1; 6 \cap 2; 3; 4) = 1; 6 \setminus \emptyset = 1; 6$ ;
5.  $S^{(8)} \neq \emptyset, T^{(8)} \neq \emptyset$ , значит,  $\begin{pmatrix} q^{(8)} \\ S^{(8)} \end{pmatrix} U_{T^{(8)}}^{(8)}$  существует.
6.  $q^{(8)} = 5$ .

В результате трансформации узла-графа № 3 имеем.

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	0	0	0	$\begin{pmatrix} s^{(8)} \\ [1;7^{(8)}] \end{pmatrix} 2, 3, 4_{[1;6^{(8)}]}^{(8)}$	$\begin{pmatrix} s^{(9)} \\ [1;7^{(9)}] \end{pmatrix} 2, 3, 5_{[6;7^{(9)}]}^{(9)}$	$\begin{pmatrix} z^{(10)} \\ [1;7^{(10)}] \end{pmatrix} 2, 3, 6_{[d^{(10)}]}^{(10)}$	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	$\begin{pmatrix} i^{(11)} \\ [d^{(11)}] \end{pmatrix} 5, 3, 2_{[1;7^{(11)}]}^{(11)}$	0	$\begin{pmatrix} z^{(12)} \\ [d^{(12)}] \end{pmatrix} 5, 3, 4_{[1;6^{(12)}]}^{(12)}$	0	$\begin{pmatrix} d^{(13)} \\ [7^{(13)}] \end{pmatrix} 5, 3, 6_{[d^{(13)}]}^{(13)}$	0
6	0	$\begin{pmatrix} i^{(14)} \\ [d;5^{(14)}] \end{pmatrix} 6, 3, 2_{[1;7^{(14)}]}^{(14)}$	0	$\begin{pmatrix} i^{(15)} \\ [d^{(15)}] \end{pmatrix} 6, 3, 4_{[d^{(15)}]}^{(15)}$	$\begin{pmatrix} d^{(16)} \\ [d^{(16)}] \end{pmatrix} 6, 3, 5_{[7^{(16)}]}^{(16)}$	0	0
7	0	0	0	0	0	0	0

Так как  $p \neq v - 1$ , продолжим трансформацию для получения гамильтоновых путей, например для узла-графа № 9, и получим.

	1	2	3	4	5	6	7
1	0	0	0	0	0	$\begin{pmatrix} \phi^{(17)} \\ [d^{(17)}] \end{pmatrix} 1, 2, 3, 5, 6_{[d^{(17)}]}^{(17)}$	$\begin{pmatrix} \phi^{(17)} \\ [d^{(17)}] \end{pmatrix} 1, 2, 3, 5, 7_{[d^{(17)}]}^{(17)}$
2	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	0	0	0	0		
6	0	0	0	0	0	0	0
7	0	0	0	0	0	$\begin{pmatrix} s^{(17)} \\ [d^{(17)}] \end{pmatrix} 7, 2, 3, 5, 6_{[d^{(17)}]}^{(17)}$	0

3.2.2. *Добавление узлов-графов.* В отличие от операции добавления вершин, операция добавления узлов-графов предполагает

добавление (поглощение) как вершин из  $S$  и  $T$ , так и подмножеств  $U$  в тройке смежных узлов-графов. При этом порядок элементов в подмножествах  $U$  сохраняется. Операция добавления будет обозначаться символом  $\parallel$ .

Пусть выбраны три узла-графа  $\binom{\varphi^{(1)}}{[s^{(1)}}U_{[r^{(1)}]}^{(1)}$ ,  $\binom{\varphi^{(2)}}{[s^{(2)}}U_{[r^{(2)}]}^{(2)}$  и  $\binom{\varphi^{(3)}}{[s^{(3)}}U_{[r^{(3)}]}^{(3)}$  графа  $G[V, E]$ .

Добавлением узлов-графов называется узел-граф  $\binom{\varphi^{(w=d)}}{[s^{(w=d)}}U_{[r^{(w=d)}]}^{(w=d)} = \binom{\varphi^{(1)}}{[s^{(1)}}U_{[r^{(1)}]}^{(1)} \parallel \binom{\varphi^{(2)}}{[s^{(2)}}U_{[r^{(2)}]}^{(2)} \parallel \binom{\varphi^{(3)}}{[s^{(3)}}U_{[r^{(3)}]}^{(3)}$ , полученный по регламенту, представленному в таблице 3.

Таблица 3. Регламент добавления узлов-графов

№ п/п	Регламент	Операции
1.	$d = \max w + 1$	+ – сложения
2.	Если $i \in T^{(1)}$ , $i \in S^{(2)}$ , $j \in T^{(2)}$ , $j \in S^{(3)}$ , то $U^{(w=d)} = U^{(1)} \parallel (i + U^{(2)} + j) \parallel U^{(3)}$	+ – добавления вершин $\parallel$ – добавления тела узла-графа
3.	$S^{(w=d)} = S^{(1)} \setminus (S^{(1)} \cap U^{(w=d)})$	$\setminus$ – разность множеств
4.	$T^{(w=d)} = T^{(3)} \setminus (T^{(3)} \cap U^{(w=d)})$	$\cap$ – пересечение множеств
Проверка условий существования узла-		
5.	графа $\binom{\varphi^{(w=d)}}{[s^{(w=d)}}U_{[r^{(w=d)}]}^{(w=d)}$ .	удаление узла-графа, если условие не выполняется
6.	Пересчет значения параметра $q$ узла-графа	

Ограничения на операцию добавления узлов-графов:

- мощность множества вершин графа  $v$  не менее 7;
- операция выполняется для узлов-графов, у которых  $S^{(1)} \neq \emptyset \wedge T^{(1)} \neq \emptyset \wedge S^{(2)} \neq \emptyset \wedge T^{(2)} \neq \emptyset \wedge S^{(3)} \neq \emptyset \wedge T^{(3)} \neq \emptyset$ , а также  $U^{(1)} \cap U^{(2)} = \emptyset$ ,  $U^{(1)} \cap U^{(3)} = \emptyset$ ,  $U^{(2)} \cap U^{(3)} = \emptyset$ .

Очевидно, что множество путей длиной  $p + 4$  получается из множества путей длины  $p$ , то есть  $k = 4$ . Добавление двух узлов-графов с одной из сторон является частным случаем добавления узлов-графов.

Алгоритм операции добавления узлов-графов проиллюстрируем на узлах-графах № 1, 6 и 5 для графа, представленного на рисунке 1:

Шаг 1. В ячейки  $f_{i,j}$  таблицы  $F$  размером  $\nu \times \nu$ , если  $i \in S^{(3)}$ ,  $j \in T^{(3)}$  и  $i \neq j$ , вписывается узел-граф  $\begin{matrix} (4^{(6)}) \\ [3;4;5^{(6)}] \end{matrix} \begin{matrix} 6^{(6)} \\ [3;4^{(6)}] \end{matrix}$  (центр).

Шаг 2. В ячейки  $m_i$  таблицы-столбца  $M$  размером  $\nu \times 1$ , если  $i \in T^{(1)}$ , вписывается узел-граф  $\begin{matrix} (4^{(1)}) \\ [2;4^{(1)}] \end{matrix} 1^{(1)}_{[2;4;7^{(1)}]}$  (периферия).

Шаг 3. В ячейки  $l_j$  таблицы-строки  $L$  размером  $1 \times \nu$ , если  $j \in S^{(7)}$ , вписывается узел-граф  $\begin{matrix} (4^{(5)}) \\ [3;7^{(5)}] \end{matrix} 5^{(5)}_{[3;6;7^{(5)}]}$  (периферия).

Шаг 4. Если ячейка  $f_{i,j}$  на пересечении строки  $i$  со столбцом  $j$  таблицы  $F$  не пуста, то выполняем регламент операции добавления узлов-графов.

В результате выполнения шагов 1–3 имеем.

		0	0	$\begin{matrix} (4^{(5)}) \\ [3;7^{(5)}] \end{matrix} 5^{(5)}_{[3;6;7^{(5)}]}$	0	0	0	$\begin{matrix} (4^{(5)}) \\ [3;7^{(5)}] \end{matrix} 5^{(5)}_{[3;6;7^{(5)}]}$
		1	2	3	4	5	6	7
	1	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0
	3	0	0	0	$\begin{matrix} (4^{(6)}) \\ [3;4;5^{(6)}] \end{matrix} 6^{(6)}_{[3;4^{(6)}]}$	0	0	0
	4	0	0	$\begin{matrix} (4^{(6)}) \\ [3;4;5^{(6)}] \end{matrix} 6^{(6)}_{[3;4^{(6)}]}$	0	0	0	0
	5	0		$\begin{matrix} (4^{(6)}) \\ [3;4;5^{(6)}] \end{matrix} 6^{(6)}_{[3;4^{(6)}]}$	$\begin{matrix} (4^{(6)}) \\ [3;4;5^{(6)}] \end{matrix} 6^{(6)}_{[3;4^{(6)}]}$	0		0
	6	0		0	0	0	0	0
	7	0	0	0	0	0	0	0

Действия на шаге 4 продемонстрируем для ячейки  $f_{4,3}$ , так как только эта ячейка удовлетворяет условиям выполнения этого шага.

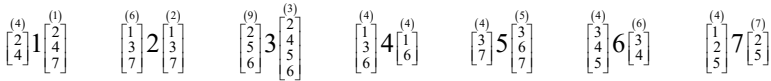
1.  $d = \max w + 1 = 7 + 1 = 8$ ;
2.  $U^{(8)} = U^{(1)} \parallel (i + U^{(6)} + j) \parallel U^{(5)} = 1 \parallel (4 + 6 + 3) \parallel 5 = 1, 4, 6, 3, 5$ ;
3.  $S^{(8)} = S^{(1)} \setminus (S^{(1)} \cap U^{(8)}) = 2; 4 \setminus (2; 4 \cap 1; 4; 6; 3; 5) = 2; 4 \setminus 4 = 2$ ;
4.  $T^{(8)} = T^{(5)} \setminus (T^{(5)} \cap U^{(8)}) = 3; 6; 7 \setminus (3; 6; 7 \cap 1; 4; 6; 3; 5) = 3; 6; 7 \setminus 6; 3 = 7$ ;

5.  $\begin{pmatrix} 1 \\ 2 \end{pmatrix}^{(8)} 1, 4, 6, 3, 5 \begin{pmatrix} 8 \\ 7 \end{pmatrix}$  существует;

6.  $q^{(8)} = 1$ .

3.2.3. *Пересечение узлов-графов.* Для пояснения операции пересечения узлов-графов элементы подмножеств  $S$  и  $T$  будут располагаться вертикально без разделительных знаков.

Тогда граф на рисунке 1 будет представляться совокупностью узлов-графов следующего вида.



Предварительные пояснения. Пусть дан узел-граф  $\begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} w \\ r \end{pmatrix}$ , у которого  $u = 2$ . Разобьем тело узла-графа на два сегмента  $U^{(w)} = \parallel_{i=1}^2 U_{\{i\}}$ . Между сегментами вставим подмножество несвязанных вершин  $R_{\{j\}}$ , через которые существуют пути длиной  $p = 2$  между  $U_{\{1\}}$  и  $U_{\{2\}}$ . Такой узел-граф имеет вид, представленный на рисунке 3.

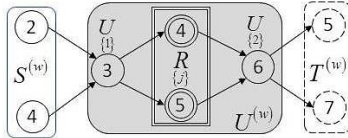


Рис. 3. Графическое изображение узла-графа

Условия существования такого узла-графа несколько расширяются:

– при  $S^{(w)} \neq \emptyset \wedge T^{(w)} \neq \emptyset \wedge R_{\{j\}} \neq \emptyset$ ;

– при  $s = t = 1$  и  $S^{(w)} \neq T^{(w)}$ , и при  $s = r = 1$  и  $S^{(w)} \neq R_{\{j\}}$ , и при

$t = r = 1$   $T^{(w)} \neq R_{\{j\}}$ , где  $r$  – мощность множества  $R_{\{j\}}$ .

Пусть имеются три узла-графа  $\begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 1 \\ r \end{pmatrix}$ ,  $\begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 2 \\ r \end{pmatrix}$  и  $\begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 3 \\ r \end{pmatrix}$ .

Пересечением узлов-графов называется узел-граф  $\begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} w=d \\ r \end{pmatrix} = \begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 1 \\ r \end{pmatrix} \cap \begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 2 \\ r \end{pmatrix} \cap \begin{pmatrix} q \\ s \end{pmatrix} U \begin{pmatrix} 3 \\ r \end{pmatrix}$ , полученный по регламенту пересечения, представленному в таблице 4.

Таблица 4. Регламент пересечения узлов-графов

№ п/п	Регламент	Операции	
1.	$d = \max w + 1$	+ – сложение	
2.	$R = \langle T^{(1)} \cap S^{(2)} \rangle \setminus U^{(3)}$	Если $u^{(1)} \geq 2$ , то вместо $U^{(1)}$ берется $\parallel_{i=u^{(1)}} U_{\{i\}}$ Если $u^{(2)} \geq 2$ , то вместо $U^{(2)}$ берется $\parallel_{i=u^{(2)}} U_{\{i\}}$ Если $u^{(3)} \geq 2$ , то вместо $U^{(3)}$ берется $\parallel_{i=u^{(3)}} U_{\{i\}}$	
3.	$R = \langle T^{(2)} \cap S^{(3)} \rangle \setminus U^{(1)}$		
4.	$S^{(w=d)} = \left[ \left\{ S^{(1)} \setminus (S^{(1)} \cap U^{(2)}) \right\} \setminus \setminus (S^{(1)} \cap U^{(3)}) \right]$		
5.	$T^{(w=d)} = \left[ \left\{ T^{(3)} \setminus (T^{(3)} \cap U^{(1)}) \right\} \setminus \setminus (T^{(3)} \cap U^{(2)}) \right]$		
6.	Если $s^{(w=d)} = 1$ , то $R = R \setminus_{\{1\}} S^{(w=d)}$ , $R = R \setminus_{\{2\}} S^{(w=d)}$ , $T^{(w=d)} = T^{(w=d)} \setminus S^{(w=d)}$		$\setminus$ – разность множеств $\cap$ – пересечение множеств $\parallel$ – добавления сегмента тела узла-графа
7.	Если $t^{(w=d)} = 1$ , то $R = R \setminus_{\{1\}} T^{(w=d)}$ , $R = R \setminus_{\{2\}} T^{(w=d)}$ , $S^{(w=d)} = S^{(w=d)} \setminus T^{(w=d)}$		
8.	Если $r = 1$ , то $R = R \setminus_{\{2\}} R_{\{1\}}$ , $S^{(w=d)} = S^{(w=d)} \setminus_{\{1\}} R$ , $T^{(w=d)} = T^{(w=d)} \setminus_{\{1\}} R$		
9.	Если $r = 1$ , то $R = R \setminus_{\{1\}} R_{\{2\}}$ , $S^{(w=d)} = S^{(w=d)} \setminus_{\{2\}} R$ , $T^{(w=d)} = T^{(w=d)} \setminus_{\{2\}} R$		
10.	$U^{(w=d)} = U_{\{1\}}, \langle R \rangle_{\{1\}}, U_{\{2\}}, \langle R \rangle_{\{2\}}, U_{\{3\}}$	, – признак смежности	
11.	Проверка условий существования узла-графа ${}^{(q)}U_{[s]}^{(w=d)}$ ${}^{(r)}$	удаление узла-графа, если условие не выполняется	
12.	Пересчет значения параметра $q$ узла-графа		

Ограничения на операцию такие же, что и для операции добавления узлов-графов.

Пересечение двух узлов-графов с любой из сторон является частным случаем пересечения узлов-графов.

Выполним операцию пересечения, например для узлов-графов под номерами 1, 6 и 5 (рисунок 1), и получим:

$$1. d = \max w + 1 = 7 + 1 = 8;$$

$$2. R_{\{1\}} = \langle T^{(1)} \cap S^{(6)} \rangle \setminus U^{(5)} = \left\langle \begin{bmatrix} 2 \\ 4 \\ 7 \end{bmatrix} \cap \begin{bmatrix} 3 \\ 4 \\ 6 \end{bmatrix} \right\rangle \setminus 5 = \langle 4 \rangle \setminus 5 = \langle 4 \rangle_{\{1\}};$$

$$3. R_{\{2\}} = \langle T^{(6)} \cap S^{(5)} \rangle \setminus U^{(1)} = \left\langle \begin{bmatrix} 3 \\ 4 \end{bmatrix} \cap \begin{bmatrix} 3 \\ 7 \end{bmatrix} \right\rangle \setminus 1 = \langle 3 \rangle \setminus 1 = \langle 3 \rangle_{\{2\}};$$

$$4. S^{(8)} = \left[ \left\{ S^{(1)} \setminus (S^{(1)} \cap U^{(6)}) \right\} \setminus (S^{(1)} \cap U^{(5)}) \right] = \left[ \left\{ \begin{bmatrix} 2 \\ 4 \end{bmatrix} \setminus \left( \begin{bmatrix} 2 \\ 4 \end{bmatrix} \cap 6 \right) \right\} \setminus \left( \begin{bmatrix} 2 \\ 4 \end{bmatrix} \cap 5 \right) \right] = \left[ \left\{ \begin{bmatrix} 2 \\ 4 \end{bmatrix} \setminus \emptyset \right\} \setminus \emptyset \right] = \begin{bmatrix} 2 \\ 4 \end{bmatrix}^{(8)};$$

$$5. T^{(8)} = \left[ \left\{ T^{(5)} \setminus (T^{(5)} \cap U^{(1)}) \right\} \setminus (T^{(5)} \cap U^{(6)}) \right] = \left[ \left\{ \left( \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \end{bmatrix} \setminus \left( \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \end{bmatrix} \cap 1 \right) \right) \setminus \left( \begin{bmatrix} 2 \\ 3 \\ 6 \\ 7 \end{bmatrix} \cap 6 \right) \right\} \setminus \emptyset \right] \setminus 6 = \begin{bmatrix} 2 \\ 3 \\ 7 \end{bmatrix}^{(8)};$$

$$8. r_{(1)} = 1, R_{\{2\}} = R_{\{2\}} \setminus R_{\{1\}} = \langle 3 \rangle \setminus \langle 4 \rangle = \langle 3 \rangle_{\{2\}}, S^{(8)} = S^{(8)} \setminus R_{\{1\}} = \begin{bmatrix} 2 \\ 4 \end{bmatrix} \setminus \langle 4 \rangle = \begin{bmatrix} 2 \end{bmatrix}^{(8)},$$

$$T^{(8)} = T^{(8)} \setminus R_{\{1\}} = \begin{bmatrix} 2 \\ 3 \\ 7 \end{bmatrix} \setminus \langle 4 \rangle = \begin{bmatrix} 2 \\ 3 \\ 7 \end{bmatrix}^{(8)};$$

$$6. s_{(1)}^{(8)} = 1, R_{\{1\}} = R_{\{1\}} \setminus S^{(8)} = \langle 4 \rangle \setminus \begin{bmatrix} 2 \end{bmatrix} = \langle 4 \rangle_{\{1\}}, R_{\{2\}} = R_{\{2\}} \setminus S^{(8)} = \langle 3 \rangle \setminus \begin{bmatrix} 2 \end{bmatrix} = \langle 3 \rangle_{\{2\}},$$

$$T^{(8)} = T^{(8)} \setminus S^{(8)} = \begin{bmatrix} 2 \\ 3 \\ 7 \end{bmatrix} \setminus \begin{bmatrix} 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 7 \end{bmatrix}^{(8)};$$

9.  $r = 1$ ,  $R = R \setminus R = \langle 4 \rangle \setminus \langle 3 \rangle = \langle 4 \rangle$ ,  $S^{(8)} = S^{(8)} \setminus R = [2] \setminus \langle 3 \rangle = [2]^{(8)}$ ,  
 $T^{(8)} = T^{(8)} \setminus R = \begin{bmatrix} 3 \\ 7 \end{bmatrix} \setminus \langle 3 \rangle = [7]^{(8)}$ ;
7.  $t^{(8)} = 1$ ,  $R = R \setminus T^{(8)} = \langle 4 \rangle \setminus [7] = \langle 4 \rangle$ ,  $R = R \setminus T^{(8)} = \langle 3 \rangle \setminus [7] = \langle 3 \rangle$ ,  
 $S^{(8)} = S^{(8)} \setminus T^{(8)} = [2] \setminus [7] = [2]^{(8)}$ ;
10.  $U^{(8)} = U_{\{1\}, \langle R \rangle}, U_{\{6\}, \langle R \rangle}, U = 1, \langle 4 \rangle, 6, \langle 3 \rangle, 5$ ;
11.  ${}_{[s]}^{(q)}U_{[r]}^{(8)} = {}_{[2]}^{(1)}1, \langle 4 \rangle, 6, \langle 3 \rangle, 5_{[7]}^{(8)}$  – существует.
12.  $q^{(8)} = 1$ .

**3.3. Предварительная оценка метода трансформации узлов-графов.** Здесь представлены соотношения, проверенные эмпирическим методом при решении очень большого числа задач на графах. Для получения верхней оценки сложности решения задачи перечисления гамильтоновых путей рассмотрен полный ориентированный граф  $G[V, E]$ . Для пересчета всех гамильтоновых путей в графе использовалась операция пересечения узлов-графов и получено следующее.

Граф задается  $v$  узлами-графами, которые объединяют пути длиной  $p = 2$ , то есть количество узлов-графов (пространство перебора) на входе задачи составит  $b_{(p=2)} = v$ . Выполнение операции пересечения позволяет перейти от путей длиной  $p = 2$  к путям длиной  $p = 6$ , потом к путям длиной  $p = 10$  и т.д. То есть минимальный размер шага  $k = 4$ , его и будем рассматривать. Для поиска всех путей в графе с каждым из узлов-графов производится операция пересечения с каждой парой узлов-графов, у которых  $U^{(1)} \cap U^{(2)} = \emptyset$ ,  $U^{(1)} \cap U^{(3)} = \emptyset$ ,  $U^{(2)} \cap U^{(3)} = \emptyset$ . При поиске путей длиной  $p = p + k$  количество таких пар составит  $\left(v - \frac{p-4}{2}\right)^2 - v + \frac{p-4}{2}$ . Из этого следует, что число узлов-графов, объединяющих пути длиной



$p = p + k$ , полученных при выполнении операции пересечения, будет составлять  $b_{(p)} = b_{(p-4)} \left[ \left( v - \frac{p-4}{2} \right)^2 - v + \frac{p-4}{2} \right]$ .

Тогда для графов с мощностью множества вершин  $v = 7, 11, 15, 19, \dots$  и  $k = 4$  имеем:

$$b_{(p)} = \begin{cases} v & \text{при } p = 2; \\ b_{(p-4)} \left[ \left( v - \frac{p-4}{2} \right)^2 - v + \frac{p-4}{2} \right] & \text{при } p = 6, 10, 14, 18, \dots \end{cases}$$

Ограничение, накладываемое на  $v$ , призвано упростить расчетные выражения, но принципиальные зависимости для графов с произвольными мощностями вершин  $v$  сохраняются.

Общее количество объектов перебора при решении задачи перечисления гамильтоновых путей составит  $B(v) = \sum_{p=2,6,10,\dots,(v-1)} b_{(p)}$ , то

есть менее чем  $0,019 \frac{v!}{\left( \left( \frac{v+1}{2} \right)! \right)}$ . Очевидно, что пространство

перебора сократилось чуть более чем в  $\left( \left( \frac{v+1}{2} \right)! \right)$  раз. Например, для полного ориентированного графа, у которого мощность множества вершин составляет  $v = 39$ , количество комбинаторных объектов, участвующих в переборе путей, уменьшается более чем в  $1,2 \cdot 10^{17}$  раз. А это очень значительное сокращение.

Нетрудно определить, что каждый узел-граф, например с мощностью  $v = 7$ , при представлении полного графа  $G[V, E]$  объединяет  $a_{(p=2)} = 30$  путей, а узел-граф, полученный после выполнения операции пересечения, объединяет  $a_{(p=6)} = 24$  гамильтонова пути. Эти результаты связываются выражением

$a_{(p)} = \frac{\left( v - \frac{p}{2} \right)!}{(v-p-1)!}$ , справедливым для полных графов мощности  $v = 7, 11, 15, 19, \dots$  и путей длины  $p = 6, 10, 14, 18, \dots$

Общее количество гамильтоновых путей в полном ориентированном графе составляет  $g = v!$ , а для графа, состоящего из 7 вершин,  $g = 5040$ .

Для этого же графа при поиске путей методом трансформации получается  $b_{(p=6)} = 210$  узлов-графов, каждый из которых объединяет  $a_{(p=6)} = 24$  гамильтонова пути. Тогда  $g = b_{(p=6)} a_{(p=6)} = 5040$ . Таким образом, метод трансформации узлов-графов позволяет точно решать задачу перечисления гамильтоновых путей в графе.

**4. Заключение.** Принципиальными особенностями предлагаемого метода трансформации узлов-графов являются следующие факторы:

1. В отличие от эвристик, которые основываются на предположениях о наиболее перспективных ветвях процесса перебора:

- для сокращения пространства перебора используется разбиение множеств  $V$  и  $E$  графа  $G[V, E]$  на более крупные комбинаторные объекты – узлы-графы;
- производится точное решение задачи.

2. В отличие от методов BFS и DFS, перебор производится путем центрально-периферического поглощения комбинаторных объектов: смежных узлов-графов, вершин и дуг.

По предварительной оценке, метод трансформации узлов-графов с размером шага  $k = 4$  позволяет значительно ускорить перебор гамильтоновых путей при сохранении точности решения задачи.

Как и любой метод перебора, метод трансформации узлов-графов обладает свойством универсальности. То есть метод трансформации узлов-графов можно использовать для разработки множества алгоритмов:

- нахождения хотя бы одного решения задачи (задача существования);
- получения всех решений задачи;
- поиска оптимального решения задачи;
- подсчета количества решений задачи.

Однако для отдельных задач анализа сложных систем, например задачи расчета структурной надежности по совокупности путей или сечений [21], потребуется некоторое развитие правил преобразования выражений символического умножения для учета пересечений путей по общим элементам.

## Литература

1. Monch C., Rizk A. Directed Acyclic Graph-Type Distributed Ledgers via Young-Age Preferential Attachment // *Stochastic Systems*. 2023. vol. 13. no. 3. pp. 377–397.
2. Chopra S., Park H., Shim S. Extended Graph Formulation for the Inequity Aversion Pricing Problem on Social Networks // *INFORMS Journal on Computing*. 2022. vol. 34. no. 3. pp. 1327–1344.
3. Vidal T., Martinelli R., Pham T.A., На М.Н. Arc Routing with Time-Dependent Travel Times and Paths // *Transportation 2021 Science*. vol. 55. no. 3. pp. 706–724.
4. Гасников А.В. Об эффективной вычислимости конкурентных равновесий в транспортно-экономических моделях // *Математическое моделирование*: 2015. Т. 27. № 12. С. 121–136.
5. Волков А.С., Баскаков А.Е. Разработка алгоритма многопутевой маршрутизации в программно-конфигурируемых сетях связи // *T-Comm: Телекоммуникации и транспорт*. 2021. Т. 15. № 9. С. 17–23.
6. Ray A., Ventresca M., Kannan K. A Graph-Based Ant Algorithm for the Winner Determination Problem in Combinatorial Auctions // *Information Systems Research*. 2021. vol. 32. no. 4. pp. 1099–1114.
7. Glasserman P., de Larrea E.L. Maximum Entropy Distributions with Applications to Graph Simulation. *Operations Research*. 2023. vol. 71. no. 5. pp. 1908–1924.
8. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи // *М. Мир*. 1982. 419 с.
9. Батенков К.А. Точные и граничные оценки вероятностей связности сетей связи на основе метода полного перебора типовых состояний // *Труды СПИИРАН*. 2019. Т. 18. № 5. С. 1093–1118.
10. Тимошенко А.В., Кочкаров Р.А., Кочкаров А.А. Выделение условий разрешимости NP-полных задач для класса предфрактальных графов // *Моделирование и анализ информационных систем*. 2021. Т. 28. № 2. С. 126–135.
11. Левин Л.А. Универсальные задачи перебора // *Проблемы передачи информации*. 1973. Т. 9. № 3. С. 115–116.
12. Gavanelli M., Mancini T. RCRA 2007: Experimental evaluation of algorithms for solving problems with combinatorial explosion. *Journal of Algorithms*. 2008. vol. 63. no. 1–3. pp. 1–2. DOI: 10.1016/j.jalgor.2008.02.002.
13. Кристофидес Н. Теория графов Алгоритмический подход // *М.: Мир*, 1978. 432 с.
14. Kudelia V.N. Full enumeration methods on graphs // *T-Comm*. 2023. vol. 17. № 7. pp. 57–64. DOI: 10.36724/2072-8735-2023-17-7-57-64.
15. Куделя В.Н. Методы перечисления путей в графе // *Научно-технические технологии в космических исследованиях Земли*. 2023. Т. 15. № 5. С. 28–38.
16. Ott F., Markovic D., Strobel A., Kiebel S.J. Dynamic integration of forward planning and heuristic preferences during multiple goal pursuit // *PLOS Computational Biology*. 2020. vol. 16. no. 2. DOI: 10.1371/journal.pcbi.1007685.
17. Banville F., Gravel D., Poisot T. What constrains food webs? A maximum entropy framework for predicting their structure with minimal biases // *PLOS Computational Biology*. 2023. vol. 19. no. 9. DOI: 10.1371/journal.pcbi.1011458.
18. Harary F., Palmer E.M. Graphical enumeration. Academic Press New York and London, 1972. 271 p.
19. Kelil A, Dubreuil B, Levy E.D, Michnick S.W. Exhaustive search of linear information encoding protein-peptide recognition. *PLOS Computational Biology*. 2017. vol. 13. no. 4. DOI: 10.1371/journal.pcbi.1005499.
20. Рыбалов А.Н. О генерической сложности проблемы распознавания гамльтоновых путей // *Прикладная дискретная математика*. 2021. № 53. С. 120–126.

21. Мизин И.А., Богатырев В.А., Кулешов А.П. Сети коммутации пакетов // М. Радио и связь. 1986. 408 с.

**Куделя Виктор Николаевич** — ведущий специалист, АО «Институт Сетевых Технологий»; профессор кафедры, кафедра интеллектуальных систем автоматизации и управления, Федеральное государственное бюджетное образовательное учреждение высшего образования «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича» (СПбГУТ). Область научных интересов: сети передачи данных, применение методов теории графов и информатики в системах маршрутизации и распределения потоков, устойчивость (надежность, живучесть) систем, алгоритмическое обеспечение автоматизированных систем. Число научных публикаций — 106. Kudelia.Viktor@int.spb.ru; 17-я линия В.О., 54-1, 199178, Санкт-Петербург, Россия; р.т.: +7(812)331-8393,,215.

V. KUDELIA

**SOLVING PATHS SEARCH PROBLEMS IN COMPLEX GRAPHS*****Kudelia V. Solving Paths Search Problems in Complex Graphs.***

**Abstract.** The construction of models of various systems is associated with the enumeration of the values of the parameters of the elements of the structure and taking into account all the characteristics of operation and interaction of components to find a certain set of solutions that determine the configuration of the system. Such tasks belong to enumeration type tasks and imply that some of the next solutions from this set are obtained from the previous solution in a certain order. It is known that any problem of the enumeration type is solved only by methods of exhaustive search, and other methods for their enumeration do not exist yet. The paper presents a new method of searching paths in a graph – the method of node-graph transformation. The proposed method, unlike the existing ones, allows one to search all directed simple paths in an oriented graph of arbitrary structure much faster. In the known graph search methods (Breadth First Search and Depth First Search), the object of the search is a path. The total number of such paths in the graph determines the size of the search space. The main idea of the node-graph transformation method is to significantly reduce the size of the search space by enlarging the search objects. The enlargement of enumeration objects is performed by clustering paths into combinatorial objects that concentrate some set of paths of the same length according to certain rules. These combinatorial objects are called node-graphs. A node-graph refers to center-peripheral combinatorial objects, and specific node-graph transformation operations have been developed to enumerate all paths in the graph, which allow finding new paths based on previous paths. The method can be used as a basic toolkit to reduce the dimensionality of the search space for solutions to NP-complete problems while maintaining the universality and accuracy of the full search.

**Keywords:** graph, enumeration, path, Hamiltonian path, combinatorial explosion, exhaustive search, NP-complete problem, clustering, node-graph.

**References**

1. Monch C., Rizk A. Directed Acyclic Graph-Type Distributed Ledgers via Young-Age Preferential Attachment. *Stochastic Systems*. 2023. vol. 13. no. 3. pp. 377–397.
2. Chopra S., Park H., Shim S. Extended Graph Formulation for the Inequity Aversion Pricing Problem on Social Networks. *INFORMS Journal on Computing*. 2022. vol. 34. no. 3. pp. 1327–1344.
3. Vidal T., Martinelli R., Pham T.A., Ha M.H. Arc Routing with Time-Dependent Travel Times and Paths. *Transportation 2021 Science*. vol. 55. no. 3. pp. 706–724.
4. Gasnikov A.V. [On the effective computability of competitive equilibria in transport-economic models]. *Matematicheskoe modelirovanie – Mathematical Modeling*: 2015. vol. 27. no. 12. pp. 121–136. (In Russ.).
5. Volkov A.S., Baskakov A.E. [Development of multipath routing algorithm in software-configurable communication networks]. *T-Comm: Telekommunikacii i transport – T-Comm: Telecommunications and Transportation*. 2021. vol. 15. no. 9. pp. 17–23. (In Russ.).
6. Ray A., Ventresca M., Kannan K. A Graph-Based Ant Algorithm for the Winner Determination Problem in Combinatorial Auctions. *Information Systems Research*. 2021. vol. 32. no. 4. pp. 1099–1114.
7. Glasserman P., de Larrea E.L. Maximum Entropy Distributions with Applications to Graph Simulation. *Operations Research*. 2023. vol. 71. no. 5. pp. 1908–1924.

8. Gjeri M., Dzhonson D. Vychislitel'nye mashiny i trudnoreshaemye zadachi [Computing machines and intractable problems]. M.: Mir. 1982. 419 p. (In Russ.).
9. Batenkov K.A. [Accurate and boundary estimate of communication network connectivity probability based on model state complete enumeration method]. Trudy SPIIRAN – SPIIRAS Proceedings. 2019. vol. 18. no. 5. pp. 1093–1118. (In Russ.).
10. Timoshenko A.V., Kochkarov R.A., Kochkarov A.A. [Allocation of solvability conditions of NP-complete problems for a class of pre-fractal graphs]. Modelirovanie i analiz informacionnyh sistem sistem – Information systems modeling and analysis. 2021. vol. 28. no. 2. pp. 126–135. (In Russ.).
11. Levin L.A. [Universal problems of brute force]. Problemy peredachi informacii – Information transfer problems. 1973. vol. 9. no. 3. pp. 115–116. (In Russ.).
12. Gavanelli M., Mancini T. RCRA 2007: Experimental evaluation of algorithms for solving problems with combinatorial explosion. Journal of Algorithms. 2008. vol. 63. no. 1–3. pp. 1–2. DOI: 10.1016/j.jalgor.2008.02.002.
13. Kristofides N. Teorija grafov Algoritmicheskij podhod [Graph Theory Algorithmic Approach]. M.: Mir, 1978. 432 p. (In Russ.).
14. Kudelia V.N. Full enumeration methods on graphs. T-Comm. 2023. vol. 17. № 7. pp. 57–64. DOI: 10.36724/2072-8735-2023-17-7-57-64.
15. Kudelia V.N. [Methods for enumerating paths in a graph]. Naukoemkie tehnologii v kosmicheskijh issledovanijah Zemli – H&ES Reserch. 2023. vol. 15. no. 5. pp. 28–38. (In Russ.).
16. Ott F., Markovic D., Strobel A., Kiebel S.J. Dynamic integration of forward planning and heuristic preferences during multiple goal pursuit. PLOS Computational Biology. 2020. vol. 16. no. 2. DOI: 10.1371/journal.pcbi.1007685.
17. Banville F., Gravel D., Poisot T. What constrains food webs? A maximum entropy framework for predicting their structure with minimal biases. PLOS Computational Biology. 2023. vol. 19. no. 9. DOI: 10.1371/journal.pcbi.1011458.
18. Harary F., Palmer E.M. Graphical enumeration. Academic Press New York and London, 1972. 271 p.
19. Kelil A., Dubreuil B, Levy E.D, Michnick S.W. Exhaustive search of linear information encoding protein-peptide recognition. PLOS Computational Biology. 2017. vol. 13. no. 4. DOI: 10.1371/journal.pcbi.1005499.
20. Rybalov A.N. [On the generic complexity of the Hamiltonian path recognition problem] Prikladnaja diskretnaja matematika – Applied Discrete Mathematics. 2021. no. 53. pp. 120–126. (In Russ.).
21. Mizin I.A., Bogatyrev V.A., Kuleshov A.P. Seti kommutacii paketov [Packet switching networks]. M.: Radio i svjaz', 1986. 408 p. (In Russ.).

**Kudelia Victor** — Leading specialist, «Institute of Network Technologies»; Professor of the department, Department of Intelligent Automation and Control Systems, Federal State Budget-Financed Educational Institution of Higher Education The Bonch-Bruевич Saint Petersburg State University of Telecommunications (SPbSUT). Research interests: data transmission networks, application of graph theory and computer science methods in routing and flow distribution systems, resilience computer network, algorithmic support of automated systems. The number of publications — 106. Kudelia.Viktor@int.spb.ru; 54-1, 17th line V.O., 199178, St. Petersburg, Russia; office phone: +7(812)331-8393,,215.