На правах рукописи

Макаровских Татьяна Анатольевна

МЕТОДЫ И АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ МАРШРУТИЗАЦИИ СПЕЦИАЛЬНОГО ВИДА В ПЛОСКИХ ГРАФАХ

05.13.17 – Теоретические основы информатики

Автореферат диссертации на соискание ученой степени доктора физико-математических наук

Челябинск – 2019

Работа выполнена на кафедре системного программирования ФГАОУ ВО «Южно-Уральский государственный университет (национальный исследовательский университет)».

Официальные оппоненты:
ЛАЗАРЕВ Александр Алексеевич, доктор физико-математических наук, профессор, заведующий лабораторией, ФГБУН «Институт проблем управления имени В.А. Трапезникова РАН», г. Москва

КАРТАК Вадим Михайлович, доктор физико-математических наук, доцент, заведующий кафедрой вычислительной техники и защиты информации, ФГБОУ ВО «Уфимский государственный авиационный технический университет», г. Уфа

МИХЕЕВА Татьяна Ивановна, доктор технических наук, профессор, профессор кафедры организации и управления перевозками на транспорте, ФГАОУ ВО «Самарский национальный исследовательский университет имени академика С.П. Королёва», г. Самара

Ведущая организация – Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б.Н. Ельцина»

Защита диссертации состоится 25 марта 2020 г., в 11:00 часов, на заседании диссертационного совета Д 212.298.18 при Федеральном государственном автономном образовательном учреждении высшего образования «Южно-Уральский государственный университет (национальный исследовательский университет)»: 454080, г. Челябинск, пр. Ленина, д. 76.

С диссертацией можно ознакомиться в библиотеке Южно-Уральского государственного университета и на сайте –
https://www.susu.ru/ru/dissertation/d-21229818/makarovskih-tatyana-anatolevna

Автореферат разослан «___» января 2020 г.

Ученый секретарь диссертационного совета

канд. физ.-мат. наук, доцент Цымблер М.Л.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы

Интерес к задачам маршрутизации объясняется их использованием в качестве математических моделей многих проблем управления и автоматизации проектирования. Многие задачи нахождения маршрутов, удовлетворяющих определенным ограничениям, появились из конкретных практических ситуаций. Всевозможные траекторные задачи являются универсальными математическими моделями многих задач оптимизации и управления: 1) эвристические алгоритмы для построения маршрутов (S.Q. Xie, Y. Jing, C. Zhige, M.K. Lee, K.B. Kwon, J. Hoeft, U.S. Palekar); 2) траекторная стабилизация мобильных роботов (ИПУ им. В.А. Трапезникова РАН, г. Москва, В.А. Уткин); 3) управление маршрутизацией и оптимизация(ИПУ им. В.А Трапезникова РАН, г. Москва, А.А. Лазарев); 4) задачи построения маршрутов в графах (H. Fleischner); 5) задача маршрутизации для вырезания заготовок из листового материала (В.А. Верхотуров, В.М. Картак, А.А. Петунин, А.Г. Ченцов, В.Д. Фроловский).

Диссертационная работа, в основном, посвящена задачам маршрутизации в плоских графах, являющихся гомеоморфными образами раскройных планов. Маршрут, покрывающий все границы вырезаемых деталей, определяет **траекторию движения режущего инструмента**. Ограничением является отсутствие пересечения внутренних граней любой начальной части маршрута с ребрами его оставшейся части. При построении систем управления манипуляторами с помощью неориентированного графа отображают всевозможные элементы траектории манипулятора. При этом возникают проблемы построения маршрутов, удовлетворяющих различным ограничениям.

Для промышленных и проектных предприятий, связанных по роду деятельности с задачами раскроя листового материала, возникает необходимость использования CAD/CAM-систем технологической подготовки процессов раскроя. Учет возможностей современного оборудования для вырезания деталей из листового материала позволяет составлять раскройные планы, допускающие совмещение контуров вырезаемых деталей, что позволяет сократить расход материала, длину резки, и количество холостых проходов. Алгоритмы составления раскройных планов для задач, допускающих совмещение, принципиально не отличаются от алгоритмов, не допускающих совмещения. Однако алгоритмы нахождения маршрутов движения режущего инструмента будут принципиально различны. Поэтому разработка алгоритмов нахождения маршрута режущего инструмента для раскройных планов, допускающих совмещение контуров вырезаемых деталей, является открытой задачей.

Степень разработанности темы исследования. На практике наибольшее распространение имеет подход, не предполагающий совмещение контуров вырезаемых деталей. Данный метод является материалоемким и энергозатратным.

Если раскройный план представляет плоский эйлеров граф, то известно, его двойственный граневый граф является бихроматическим. Для этого случая С.Б. Белым[1] доказано существование эйлерова цикла, гомеоморфного плоской жордановой кривой без самопересечений. Однако осталась не ясна возможность использования данного результата в CAD/CAM системах технологической подготовки процессов раскроя. Алгоритм, предложенный в этой работе, является полиномиальным. Впоследствии У. Манбером дано доказательство $\mathcal{NP}$-полноты данной задачи[2]. В дальнейшем Г. Фляйшнер [3] ввел понятие A-цепи, в которой разрешенные переходы между ребрами заданы циклическим порядком в каждой вершине графа. Им же было показано, что задача определения A-цепи $\mathcal{NP}$-трудна в общем случае, были приведены частные случаи, для которых задача разрешима за полиномиальное время. Одним из таких частных случаев является 4-регулярный граф. Попытки построить маршруты, в которых пройденная часть не охватывает еще непройденных ребер были предприняты в работе У. Манбера[4]. Строгая формализация указанной проблемы в терминах OE-цепей дана в работе автора [1], однако, OE-цепь допускает возможность самопересечения траектории.

Цели и задачи исследования. *Целью* диссертационного исследования является решение проблемы маршрутизации специального вида

[1] *Белый С.Б.* О самонепересекающихся и непересекающихся цепях // Математические заметки, 1983. Т. 34. № 4. С. 625–628.

[2] *Manber U., Bent S.W.* On Non-intersecting Eulerian Circuits// Discrete Applied Mathematics, Vol. 18, 1987. P. 87–94.

[3] *Фляйшнер Г.* Эйлеровы графы и смежные вопросы. М.: Мир, 2002. 335 с., ил.

[4] *Manber U., Israni S.* Pierce Point Minimization and Optimal Torch Path Determination in Flame Cutting// Journal of Manufacturing Systems, Vol. 3, No.1, 1984. P. 81–89.

в плоских графах, являющихся гомеоморфными образами раскройного плана.

Для достижения поставленной цели были поставлены и решались следующие *задачи*:

- определить способ представления гомеоморфного образа раскройного плана, позволяющего эффективно решать проблемы маршрутизации;
- доказать существование маршрутов, удовлетворяющих приведенному выше набору ограничений, для плоских графов;
- разработать методы и алгоритмы решения проблемы построения маршрутов специального вида в плоских графах и доказать их корректность;
- разработать программное обеспечение для реализации представленных алгоритмов;
- получить оценки количества полученных маршрутов специального вида.

Научная новизна полученных в диссертации результатов заключается в формировании общего подхода к решению задач маршрутизации специального вида в плоских графах и состоит в следующем.

- Введен класс OE-маршрутов в плоских графах. Маршруты введенного класса представляют упорядоченную последовательность реберно-непересекающихся цепей, покрывающих граф, в которой отсутствуют пересечения внутренности пройденной части маршрута с ребрами его непройденной части.
- Показано, что на мощность покрытия существенное влияние оказывает наличие мостов в графе. При их отсутствии минимальное число OE-цепей, покрывающих граф, совпадает с минимальным числом цепей, покрывающих данный граф (в случае существования вершин нечетной степени, инцидентных внешней грани). Если вершин нечетной степени, смежных внешней грани, нет, то мощность покрытия на единицу выше минимального числа цепей, покрывающих данный граф. В общем случае для мощности OE-покрытия графа G имеет место неравенство $k = \frac{|V_{odd}(G)|}{2} \leq N \leq |V_{odd}(G)| = 2k$. Как верхняя, так и нижняя границы достижимы.

- Доказано, что плоские эйлеровы графы имеют эйлеровы циклы, принадлежащие классу OE.
- Введен класс AOE-маршрутов в плоских графах. Маршруты введенного класса являются OE-маршрутами, для которых выполнено дополнительное локальное ограничение: следующее ребро определяется заданным циклическим порядком на множестве ребер, инцидентных текущей вершине (т.е все цепи построенного маршрута являются A-цепями).
- Введен класс NOE-маршрутов в плоских графах. Этот класс является расширением класса AOE и в него входят все маршруты, состоящие из непересекающихся OE-цепей.
- Разработаны эффективные алгоритмы нахождения в плоском графе $G = (V, E)$ маршрутов введенных классов, имеющие полиномиальную вычислительную сложность. Данные алгоритмы позволяют минимизировать длину дополнительных переходов между концевыми вершинами цепей графа $G = (V, E)$.
- Определены оценки количества OE-цепей для фиксированной системы переходов (последовательности обхода ребер).

Теоретическая ценность. Дано доказательство полиномиальной разрешимости задачи построения непересекающихся маршрутов в плоских графах и разработаны алгоритмы решения данной задачи. Полученные теоретические результаты расширяют класс задач построения маршрутов специального вида в графах и являются продолжением исследований Г. Фляйшнера, М.А. Верхотурова, Э.А. Мухачевой, А.А. Петунина и позволяют решать задачу построения маршрутов, удовлетворяющих технологическим ограничениям.

Практическая ценность. Разработанные алгоритмы могут быть применены в CAD/CAM-системах технологической подготовки процессов раскроя, ориентированных на использование ресурсосберегающих технологий. Предложенная теория дает новый импульс для построения новых методов решения задач вырезания деталей, позволяя формализовать требования к раскройным планам, ориентированным на применение ресурсосберегающих технологий.

Методы исследования. В диссертационной работе для решения задачи маршрутизации в графе, полученном в результате абстрагирования раскройного плана использован современный аппарат теории графов.

Области исследования соответствуют паспорту специальности 05.13.17 – Теоретические основы информатики, при этом работа соответствует пункту 10 паспорта специальности:

- П10. Разработка основ математической теории языков и грамматик, теории конечных автоматов и теории графов.

Достоверность результатов, полученных в диссертационной работе, базируется на использовании апробированных научных положений, методов исследования, корректном применени математического аппарата, согласовании новых научных результатов с известными теоретическими положениями. Новизна и результативность предложенных алгоритмов подтверждены свидетельствами о регистрации программ. Разработанное программное обеспечение позволяет получить решения поставленной задачи для произвольного плоского графа, соответствующего заданному на листе раскройному плану. Правообладателем разработанного в рамках данной модели комплекса программ является ФГАОУ ВО ЮУрГУ (НИУ).

Апробация работы. Все результаты диссертационной работы, разработанные методы, алгоритмы и результаты вычислительных экспериментов докладывались и получили одобрение более, чем на 50 международных, всероссийских и региональных конференциях, в том числе:

1. International Conference on Mathematical Optimization Theory and Operations Research (Ekaterinburg, July 8–12, 2019).
2. 18th Conference on Sheet Metal (Leuven, Belgium, April 15–17, 2019).
3. The 7th International Conference on Optimization Problems and Their Applications (Омск, 8–14 июля 2018 г.)
4. The 20th World Congress of the International Federation of Automatic Control (Toulouse, France, July 9–14, 2017).
5. MIM Conference, Manufacturing Modelling, Management and Control (2016, 2019).
6. Czech-Slovak International Symposium on Graph Theory, Combinatorics, Algorithms and Applications (2006, 2013).

7. 8th French Combinatorial Conference, Orsay, France (June,28–July,2, 2010).
8. 13-th IFAC Symposium on Information Control Problems in Manufacturing, Moscow, 2009.
9. International meeting «Euler and Modern Combinatorics», St. Petersburg, June 1–7, 2007.
10. 6-th International Congress on Industrial and Applied Mathematics, Zurich, 16–20 July, 2007.
11. Международная научно-техническая конференция "Перспективные информационные технологии (ПИТ-2018)"(Самара, Самарский национальный исследовательский университет имени академика С.П. Королева, 16–19 апреля, 2018).
12. Международная конференция «Системы проектирования технологической подготовки производства и управления этапами жизненного цикла промышленного продукта(CAD/CAM/PDM)» (Москва, 2015–2016, 2018 гг.).
13. Международная научно-техническая конференция «Пром-Инжиниринг» (Челябинск, 2015, 2016, 2018).
14. International Conference «Information Technologies for Intelligent Decision Making Support» (Уфа, 2013, 2015–2017).
15. 2nd International conference "Intelligent Technologies for Information Processing and Management" (ITIPM-2014, Уфа, 10–12 ноября, 2014).
16. 5th International conference "Optimization and Applications" (Optima-2014, Petrovac, Montenegro, Sep. 27–Oct.4, 2014).
17. Workshop on Computer Science and Information Technologies (2003, 2008, 2010, 2011, 2013).
18. Международная научная конференция «Дискретная математика, алгебра и их приложения», Минск, Институт математики НАН Беларуси, (2009 и 2015).
19. Российская конференция «Дискретный анализ и исследование операций», Новосибирск, Институт математики им. С.Л. Соболева СО РАН, 2002, 2004);
20. Международная конференция «Дискретная оптимизация и исследование операций» (2010, 2013, 2016).

21. Международная конференция «Информационные технологии и системы», респ. Башкортостан, оз. Банное, (2012–2017).
22. Международный семинар «Дискретная математика и ее приложения» (Москва, МГУ им. М.В. Ломоносова, 2001, 2004, 2010, 2016).

Кроме того, результаты работы были представлены на ежегодных научно-практических конференциях Южно-Уральского государственного университета.

Публикации. По теме диссертационного исследования опубликовано 101 работа, из них 6 свидетельств о государственной регистрации программ для ЭВМ, 14 статей из перечня ВАК ведущих рецензируемых изданий (работы [1–14]), 13 статей, индексируемых в SCOPUS (работы [2, 8, 11, 15–24]), 1 монография [25].

Подготовка к публикации полученных результатов проводилась совместно с соавторами, причем вклад диссертанта был определяющим. Все представленные в диссертации результаты получены лично автором. В работе [6] И. О. Алферову принадлежит описание примера на с. 55-58, Т. А. Макаровских принадлежит весь остальной текст статьи. В работе [7] Е.А. Савицкому принадлежит описание программного обеспечения в разделе 3, Т.А. Макаровских – все остальные разделы. В работе [9] Е.А. Савицкому принадлежат описания примеров раскройных планов, изображенных на рисунках 1–7, 9–10, Т.А. Макаровских принадлежат все остальные части статьи. В работе [11] Е.А. Савицкому приндлежат описания примеров кодирования графа на рисунках 1–3, 5 и в таблицах 1–2, А.В. Панюкову – введения и заключения, Т.А. Макаровских принадлежит весь остальной текст статьи. В работе [16] Э.А. Мухачевой принадлежит обзор известных методов составления раскройных планов (раздел 2), Т.А. Макаровских – все остальные разделы. В работе [17] Е.А. Савицкому принадлежит обзор алгоритмов в разделе 2, Т.А. Макаровских – остальной текст статьи. В работе [18] А.В. Панюкову принадлежит введение, Е.А. Савицкому – описание примеров раскройных планов, приведенных на рисунках 1–6, Т.А. Макаровских – остальной текст статьи. В работах [19, 22–24] А.В. Панюкову принадлежат введения и заключения, Т.А. Макаровских – все остальные разделы. В работе [24] Е.А. Савицкому принадлежат описания примеров работы алгоритмов и кодирования графа. В работе [21] Е.А. Савицкому принадлежит описание ограничений со стороны технологического процесса, Т.А. Макаровских – остальной текст статьи.

Структура работы и объем работы. Диссертация состоит из введения, 4 глав, заключения и списка литературы. Объем диссертации составляет 225 страниц. Список литературы содержит 200 наименований.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обосновывается актуальность темы диссертации, выделяются и формулируются цели и задачи исследования, описывается структурно-логическая схема диссертационной работы.

В первой главе «Современное состояние исследований задач маршрутизации» на основе аналитического обзора литературы, отражающего состояние проблемы применения графов в математическом моделировании, показано место решаемой в данной работе задачи относительно ранее опубликованных в научной литературе результатов. С тем, чтобы более четко очертить круг решаемых в данной работе задач и показать их место в общей теории графов, приведено краткое описание постановки задачи нахождения эйлеровых (обход по всем ребрам ровно по одному разу с возвратом в исходную вершину) маршрутов, указаны известные алгоритмы построения эйлеровых цепей и отмечено, что эти алгоритмы позволяют построить эйлеровы цепи или покрытия цепями без учета ограничений на порядок обхода ребер.

Граф G, как правило, имеет много эйлеровых цепей, а, следовательно, и разложений на цепи, поэтому имеется потенциальная возможность построения разложений, удовлетворяющих дополнительным ограничениям.

Практика требует построения маршрутов, удовлетворяющих различным ограничениям на последовательность ребер. В частности, ограничения можно классифицировать на:

- локальные, когда следующее ребро в маршруте определяется условиями, заданными в текущей вершине или на текущем ребре (например, исключение запрещенных переходов);
- глобальные (отсутствие пересечения внутренности пройденной части плоского графа с ребрами его непройденной части и т.д.).

Анализ публикаций показал, что большинство работ посвящено алгоритмам с локальными ограничениями на порядок обхода ребер (например, запрещение левых поворотов; маршруты без поворотов; использование в каждой вершине графа заданного циклического порядка включения ребер в маршрут и т.п.). Обобщение большинства частных случаев задачи построения маршрутов с локальными ограничениями дано С. Зейдером. Им предложено представлять локальные ограничения в каждой вершине v исходного графа G в виде грфа $G_{E(v)}$ возможных переходов. Множеством вершин графа $G_{E(v)}$ являются все ребра, инцидентные вершине v; смежные вершины графа $G_{E(v)}$ соответствуют разрешенным переходам.

Задача построения эйлеровой совместимой цепи оставалась открытой. Ее решение приведено в диссертационном исследовании.

Остается открытой задача построения эйлерова цикла, удовлетворяющего заданным локальным и глобальным ограничениям в плоском графе. В известных работах, посвященных задаче построения в плоском эйлеровом графе эйлерова цикла при наличии ограничений, отсутствуют строгая формализация и доказательство результативности алгоритма. Кроме того, известные алгоритмы имеют достаточно высокую вычислительную сложность.

Актуальной задачей является алгоритм построения плоской самонепересекающейся цепи с упорядоченным охватыванием.

Имеется потенциальная возможность построения покрытий, удовлетворяющих требуемым ограничениям, в частности, для задач маршрутизации в плоских графах. Однако, эффективных алгоритмов решения таких задач не было найдено.

Во второй главе «Маршруты с упорядоченным охватыванием» поставлена и решена задача построения в плоском графе маршрутов, удовлетворяющих условию отсутствия пересечения внутренних граней любой его начальной части с ребрами его оставшейся части. Формально такие маршруты определены как упорядоченная последовательность OE-цепей графа $G = (V, E)$ и образуют класс OE-маршрутов. Введены определения и обозначения теории маршрутов с упорядоченным охватыванием (OE-маршрутов).

Рассмотрим плоскость S, на которой задан плоский граф G с внешней гранью f_0. Для любой части графа $J \subseteq G$ обозначим через $\mathrm{Int}\,(J)$ теоретико-множественное объединение его внутренних граней (объединение всех связных компонент $S \setminus J$, не содержащих внешней грани). Тогда $\mathrm{Int}(J)$ можно интерпретировать как отрезанную от листа часть. Множества вершин, ребер и граней графа J будем обозначать через $V(J)$, $E(J)$ и $F(J)$ соответственно, а через $|M|$ – число элементов множества M.

Определение 1. *Цепь $C = v_1e_1v_2e_2\ldots v_k$ в плоском графе G имеет* ***упорядоченное охватывание*** *(является OE-цепью), если для любой его начальной части $C_l = v_1e_1v_2e_2\ldots e_l$, $l \leq (|E|)$ выполнено условие $\mathrm{Int}\,(C_l) \cap E = \emptyset$.*

Теорема 1. *Пусть G – плоский эйлеров граф. Для любой вершины $v \in V(G)$, инцидентной границе внешней (бесконечной) грани графа G, существует эйлеров OE-цикл $C = ve_1v_1e_2v_2\ldots v_{|E|-1}e_{|E|}v$.*

Определение 2. *Упорядоченная последовательность реберно-непересекающихся OE-цепей*

$$\begin{aligned} C^0 &= v^0e_1^0v_1^0e_2^0...e_{k_0}^0v_{k_0}^0, \quad C^1 = v^1e_1^1v_1^1e_2^1...e_{k_1}^1v_{k_1}^1, \ldots, \\ C^{n-1} &= v^{n-1}e_1^{n-1}v_1^{n-1}e_2^{n-1}...e_{k_{n-1}}^{n-1}v_{k_{n-1}}^{n-1}, \end{aligned}$$

покрывающая граф G и такая, что

$$(\forall m:\, m < n)\,, \quad \left(\bigcup_{l=0}^{m-1} \mathrm{Int}(C^l)\right) \cap \left(\bigcup_{l=m}^{n-1} C^l\right) = \emptyset$$

называется ***маршрутом с упорядоченным охватыванием*** *(OE-маршрутом).*

Построение OE-маршрута графа G решает поставленную задачу раскроя. Наибольший интерес представляют маршруты с минимальным числом цепей, поскольку переход от одной цепи к другой соответствует холостому проходу режущего инструмента.

Определение 3. *Маршрут, содержащий минимальную по мощности упорядоченную последовательность реберно-непересекающихся OE-цепей в плоском графе G будем называть* ***эйлеровым маршрутом с упорядоченным охватыванием*** *(эйлеровым OE-маршрутом), а составляющие его OE-цепи –* ***эйлеровым OE-покрытием****.*

Теорема 2. *Пусть G плоский связный граф, $V_{odd}(G)$ – множество вершин нечетной степени графа G, тогда для мощности N эйлерова OE-*

покрытия графа G имеет место неравенство

$$k = \frac{|V_{odd}(G)|}{2} \leq N \leq |V_{odd}(G)| = 2k.$$

Верхняя и нижняя границы достижимы.

На мощность покрытия существенное влияние оказывает наличие мостов в графе. При их отсутствии достигается нижняя граница, в случае существования вершин нечетной степени, инцидентных внешней грани; либо, если таких вершин нет, мощность покрытия на единицу выше нижней границы.

Теорема 3. *Пусть $G = (V, E)$ – плоский связный граф на S, не имеющий мостов. Существует множество ребер $H \ : \ (H \cap S) \backslash V = \emptyset$ такое, что граф $\hat{G} = (V, E \cup H)$ – эйлеров, то в графе $\hat{G}$ существует эйлеров цикл $C = v_1 e_1 v_2 e_2 ... e_n v_1$, $n = |E| + |H|$, для любой начальной части которого $C_l = v_1 e_1 v_2 e_2 ... v_l$, $l \leq |E| + |H|$, выполнено условие $\mathsf{Int}(C_l) \cap G = \emptyset$.*

В описании алгоритмов используется понятие **ранга** ребра e.

Определение 4. *Рангом ребра $e \in E(G)$ будем называть значение функции $\mathrm{rank}(e) : \ E(G) \to \mathbb{N}$, определяемую рекурсивно:*

- *пусть $E_1 = \{e \in E : \ e \subset f_0\}$ – множество ребер, ограничивающих внешнюю грань f_0 графа $G(V, E)$, тогда $(\forall e \in E_1) (\mathrm{rank}(e) = 1)$;*
- *пусть $E_k(G)$ – множество ребер ранга 1 графа*

$$G_k \left(V, E \backslash \left(\bigcup_{l=1}^{k-1} E_l \right) \right),$$

тогда $(\forall e \in E_k) (\mathrm{rank}(e) = k)$.

Топологическое представление плоского графа $G = (V, E)$ на плоскости S с точностью до гомеоморфизма определяется заданием для каждого ребра $e \in E$ следующих функций (рис. 1):

- $v_k(e)$, $k = 1, 2$ – вершины, инцидентные ребру e,
- $l_k(e)$, $k = 1, 2$ – ребра, полученные вращением ребра e против часовой стрелки вокруг вершины $v(k)$,

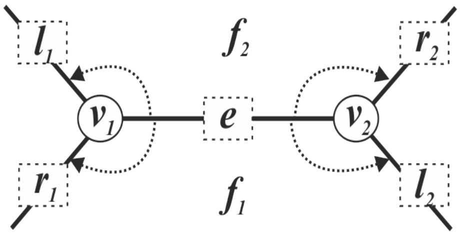


Рис. 1. Функции для представления плоского графа

- $r_k(e)$, $k = 1, 2$ – ребра, полученные вращением ребра e по часовой стрелке вокруг вершины $v(k)$,
- $f_k(e)$ – грань, находящаяся слева при движении по ребру e от вершины $v_k(e)$ к вершине $v_{3-k}(e)$, $k = 1, 2$.

Разработаны алгоритмы, которые за полиномиальное время решают задачу построения OE-маршрутов для плоского графа:

- эйлеров OE-цикл в плоском эйлеровом графе (алгоритм `OE-CYCLE`, сложность $O(|V|^2)$);
- связный OE-маршрут китайского почтальона в *произвольном* плоском связном графе; при удалении повторно проходимых ребер будет получен OE-маршрут; данный маршрут не является оптимальным ни по количеству покрывающих цепей, ни по длине холостых переходов (алгоритм `CPP_OE`, сложность алгоритма – $O(|E(G)|\cdot|V(G)|)$);
- маршрут в плоском связном графе *без мостов*, представляющий OE-покрытие, оптимальное по числу цепей, длина холостых переходов может не быть оптимальной (алгоритм `OECover` (см. алг. 1), сложность алгоритма – $O\left(|E| \cdot \log |V|\right)$);
- OE-маршрут в плоском связном графе без мостов с заданными дополнительными ребрами, соединяющими вершины нечетной степени (алгоритм `M-COVER` со сложностью $O\left(|E| \cdot \log |V|\right)$; алгоритм `M-COVER`, дополненный алгоритмом нахождения кратчайшего паросочетания, позволяет построить OE-покрытие с минимальной суммарной длиной дополнительных ребер (сложность алгоритма – $O(|E| \cdot \sqrt{|V|})$).

Алгоритм `OECover` (см. алг. 1) строит покрытие плоского графа G последовательностью OE-цепей. Как и прежде, граф G представлен списком ребер с заданными на них функциями $v_k(e)$, $l_k(e)$, $f_k(e)$, $k = 1, 2$.

Algorithm 1 Алгоритм OECover

Require: $G = (V, E)$ – плоский граф; $V_{odd} \subseteq V$ – множество вершин нечетной степени;
Ensure: $first \in E$, $last \in E$, $\mathrm{mark}_1 : E \to E$;

```
Initiate();
Order();
SortOdd();        ▷ Сортировка списка вершин нечетной степени по убыванию ранга
if {∃v ∈ V_odd| v ∈ f_0} then
    v^0 ← arg max_{v∈V_odd} rank(v);
    V_odd ← V_odd\{v^0};
else
    v^0 ← v| v ∈ f_0;
end if
do
    v ← FormChain(v^0);
    V_odd ← V_odd\{v};
    if (V_odd = ∅)
        break;
    end if
    v^0 ← arg max_{v∈V_odd} rank(v);
while(true);
end
```

В теле процедуры `Initiate` присваиваются начальные значения всех используемых переменных, а также определяется ребро $e_0 \in E$, принадлежащее границе внешней грани f_0.

Функциональное назначение процедуры `Order` состоит в:

1) определении на каждом ребре $e \in E$ значения $\mathrm{rank}(e)$ (заметим, что ранг любого ребра плоского графа может быть определен за время $O(|E|)$ с помощью данной процедуры;

2) в формировании для каждой вершины списка $Q(v)$ инцидентных ребер, упорядоченных в порядке убывания значения $\mathrm{rank}()$.

После выполнения процедур `Initiate` и `Order` выполняется упорядочение вершин нечетной степени $v \in V_{odd}$ в порядке возрастания их ранга с помощью процедуры `SortOdd`. За ранг вершины v принимается значение функции $\mathrm{rank}(Stack(v))$. Далее выполняется цикл `do...while` с использованием процедуры `FormChain` (см. алг.2). В данном цикле строится последовательность из $|V_{odd}|/2$ простых цепей между парами вершин нечетной степени. Если ни одна из вершин нечетной степени не смежна внешней грани, то необходимо построить $|V_{odd}|/2 + 1$ цепь, где первая из построенных цепей C^0 начинается в вершине четной степени v^0, смежной внешней грани, и заканчивается в вершине нечетной степе-

Algorithm 2 Процедура FormChain

procedure FormChain(In: w – начальная вершина цепи; Out: v – конечная вершина цепи)
  $v \leftarrow w$; $e \leftarrow Q(v)$;
  do
  $e_1 = \arg\max_{e \in Q(v)} \operatorname{rank}(e)$;
  $e_2 = \arg\max_{e \in Q(v): f_1(e) = f_2(e)} \operatorname{rank}(e)$;
  if $\operatorname{rank}(e_1) = \operatorname{rank}(e_2)$ **then** ▷ Найти ребро максимального ранга, по возможности не являющееся мостом
    $e = e_2$;
  else
    $e = e_1$;
  end if
  if $v = v_1(e)$ **then**
    REPLACE(e); ▷ Изменить индексы функций ребра e с k на $3-k$, $k = 1, 2$
  end if
  $E(G) \leftarrow E(G) \setminus \{e\}$; ▷ Удалить ребро e и удалить грани, разделенные ребром e
  $Trail \leftarrow Trail \cup \{e\}$;
  $v \leftarrow v_1(e)$;
  while $(v \notin V_{odd} \& Q(v) \neq \emptyset)$;
  return v;
end procedure

ни, цепи $C^1, \dots C^{n-1}$ соединяют вершины нечетных степеней, а цепь C^n начинается в вершине нечетной степени, а заканчивается в вершине v^0.

Функциональное назначение процедуры `FormChain` состоит в формировании OE-цепи, начинающейся в заданной вершине w и заканчивающейся в некоторой вершине $v \in V_{odd}$, $v \neq w$. В результате выполнения процедуры будет построена простая цепь $C^i = v_0^i e_1^i v_1^i e_2^i \dots e_k^i v_k^i$, в которой $v_1^i, v_2^i, \dots v_{k-1} \notin V_{odd}^i$, а для $i \neq 0$ и $i \neq n$ вершины $v_0^i, v_k^i \in V_{odd}$, при $i = 0$ вершина $v_k^i \in V_{odd}$, а при $i = n$ вершина $v_0^i \in V_{odd}$,

$$e_i = \arg \max_{e \in E(v_i) \setminus \{e_l | l < i\}} \operatorname{rank}(e),\ v_{i+1} = \overline{v_1}(e_i),\ i = 1, 2, \dots, k,$$

кроме того, для любой начальной части $C_l = v^0 e_1 v_1 e_2 v_2 \dots e_l$, $l \leq k$ и для любой вершины $v \in V$ имеет место неравенство

$$\min_{e \in E(v) \bigcap E(C_l)} \operatorname{rank}(e) > \max_{e \in E(v) \setminus E(C_l)} \operatorname{rank}(e).$$

Теорема 4. *Пусть $G = (V, E)$ – плоский связный граф на S, не имеющий мостов, $V_{odd} \subset V$ – множество вершин нечетной степени. Тогда для любого паросочетания M на множестве V_{Odd} в графе $\hat{G} = (V, E \cup M)$ существует эйлеров цикл $C = v_1 e_1 v_2 e_2 \dots e_n v_1$, $n = |E| + |M|$,* *для любой начальной части которого* $C_l = v_1 e_1 v_2 e_2 ... v_l$, $l \leq |E| + |M|$, *выполнено условие* $\mathsf{Int}(C_l) \cap G = \emptyset$.

Доказательство теоремы 4 конструктивно и состоит в доказательстве результативности алгоритма M-`Cover` (см. алг. 3) построения покрытия для любого паросочетания на множестве вершин нечетной степени.

Algorithm 3 Алгоритм M-`Cover`

Require: связный плоский граф G, функции $v_k(e)$, $l_k(e)$, $e \in E(G)$, $k = 1, 2$; вершина $v_0 \in V(G)$, инцидентная внешней грани; паросочетание M на множестве вершин V_{Odd} нечетной степени; булева функция $\mathtt{Idle}_M : V_{Odd} \to \{\mathbf{false}, \mathbf{true}\}$ на множестве V_{Odd} вершин нечетной степени;

Ensure: вполне упорядоченное множество C из OE-цепей графа G, представляющее OE покрытие графа G;

```
Order (G);                      ▷ Определить rank() для всех e ∈ E(G), v ∈ V(G)
v := v0;                                                      ▷ Построение
while Q(v) ≠ ∅ do
    FormChain(v, v);
    if Idle_M(v) ∨ (Q(v) = ∅) then
        u ← M(v);      ▷ Вершина u является напарником вершины v в паросочетании M
        V_Odd ← V_Odd \ {u, v}                ▷ Удалить вершины u, v из V_Odd
        v ← u;                          ▷ Завершить построение текущей цепи
    end if
end while
End of algorithm
```

Основным отличием данного алгоритма от алгоритма OE-Cover является то, что для каждой вершины $v \in V_{Odd}$ фиксирована следующая вершина $u = M(v) \in V_{Odd}$, в которую осуществляется переход. Алгоритм `M-Cover` может завершить построение текущей цепи как при первом посещении вершины $v \in V_{Odd}$, так и в тот момент, когда вершина станет тупиковой (т.е. $Q(v) = \emptyset$). Чтобы определить, в какой момент следует прервать построение цепи, используются значения

$$\mathtt{Idle}_M(v) = (\mathrm{rank}(v) \leq \mathrm{rank}(M(v))) \wedge \left(f_{M(v)} \succeq f_v\right),\ v \in V_{Odd},$$

где $f_w = \arg\min_{f: v \in f \subset F(G)} \mathrm{rank}(f), w \in V_{Odd}$. Здесь $\succeq$ – отношение частичного порядка на $F(G)$, индуцируемое деревом $T_{f0}^{G\prime}$ кратчайших путей до вершины $f_0 \in F$:

$$(f_i \succeq f_j) \leftrightarrow \left(f_j \text{ принадлежит цепи } T_{f_0}^{G\prime} \text{ между } f_i \text{и} f_0\right).$$

Для построения оптимального покрытия (т.е. покрытия с минимальной длиной дополнительных построений) достаточно в качестве M взять кратчайшее паросочетание на множестве V_{odd}. Это реализовано в следующем алгоритме.

Алгоритм `OptimalCover`

Входные данные:

- плоский граф G, представленный списком ребер с заданными на них функциями $v_k(e)$, $l_k(e)$, $f_k(e)$, $k = 1, 2$.

Выходные данные:

- C_j, $j = 1, ..., |V_{odd}|/2$, – покрытие графа G OE-цепями.

Шаг 1. Найти кратчайшее паросочетание M на множестве V_{odd}.

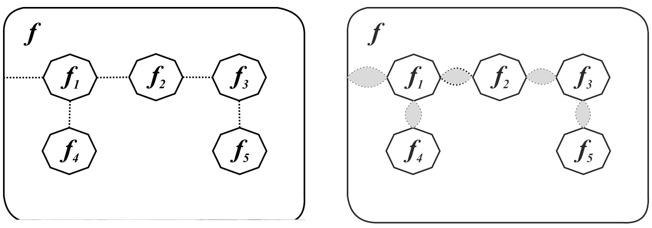
Шаг 2. Выполнить алгоритм `M-Cover` для графа G и паросочетания M.

Шаг 3. Останов.

Очевидно,что алгоритм `OptimalCover` строит оптимальное OE-покрытие, его вычислительная сложность не превосходит $O(|E(G)| \cdot \sqrt{|V(G)|})$. Данная оценка определяется вычислительной сложностью **Шага 1**.

Практическую ценность представляет задача построения OE-маршрутов в несвязных графах. В этом случае задачу поиска OE-покрытия графа цепями можно свести к ряду задач меньшей размерности: строить покрытие для каждой компоненты связности в отдельности. Если полученный граф не содержит вложенных компонент, то данный подход является разумным. Однако при наличии вложенности компонент связности задача несколько усложняется и возникают следующие ограничения на порядок обхода компонент связности: компоненты связности, состоящие из ребер более высокого ранга необходимо обходить раньше компонент, состоящих из ребер более низкого ранга.

Для решения задачи в общем случае (для плоского несвязного графа) разработаны алгоритмы `MultiComponent` (сложность алгоритма – $O(|E(G)| \cdot \log_2 |V(G)|)$), `Bridging` и `DoubleBridging`) (рис. 2). Доказа-



(a) Bridging (b) DoubleBridging

Рис. 2. Примеры объединения разделенных компонент тельства данных результатов конструктивны и фактически сводятся к описанию и доказательству результативности алгоритмов построения искомых циклов (маршрутов).

Алгоритмы `Bridging` и `DoubleBridging` используют подход, который состоит в преобразовании несвязного графа в связный.

Определение 5. *Грань $f \in F(G)$ будем называть* ***разделяющей****, если она инцидентна двум и более компонентам связности.*

Пусть граф $\tilde{G}$ получен из графа G добавлением в разделяющих гранях мостов между компонентами связности. Очевидно, что полученный граф $\tilde{G}$ будет плоским связным графом и для него может быть построен OE-маршрут $R(\tilde{G})$. Искомый OE-маршрут $R(G)$ может быть получен из маршрута $R(\tilde{G})$ если вершины инцидентные введенным мостам считать окончанием текущей цепи и началом следующей (т.е. введенные мосты считать холостыми перемещениями).

Рассмотрим способ построения мостов, связывающих граф G и имеющих минимальную суммарную длину (см. алгоритм 4).

Algorithm 4 `Bridging`

Require: плоский несвязный граф G
Ensure: плоский связный граф $\tilde{G}$ и множество B добавленных мостов
1: $\tilde{G} := G$; $B = \emptyset$;
2: Определить множество C_F всех разделяющих граней.
3: **for all** $f \in C_F$ **do**
4: Найти множество $S(f)$ компонент связности графа G, инцидентных грани f.
5: Построить полный абстрактный граф $\mathcal{T}$, вершинами которого являются компоненты связности $S(f)$, а длины ребер равны расстоянию между соответствующими компонентами.
6: Найти остовное дерево минимального веса $T(\mathcal{T})$ в $\mathcal{T}$.
7: Добавить ребра найденного остовного дерева в граф $\tilde{G}$: $E(\tilde{G} := E(\tilde{G}) \cup E(T(\mathcal{T}))$, $B := B \cup E(T(\mathcal{T}))$.
8: **end for**
9: **end**

Плоский граф $\tilde{G}$, построенный алгоритмом `Bridging`, содержит мосты, поэтому к нему можно применить только алгоритм `CPP_OE`, использующий построение маршрута китайского почтальона.

Отметим, что как для применения алгоритма `OECover`, так и для алгоритма M-`Cover` требуется отсутствие мостов в графе. Для избежания ошибок в выполнении алгоритмов необходимо дважды добавить в граф ребра остовного дерева $T(\mathcal{T})$ (см. строку 6 алгоритма 4). Скорректированный таким образом алгоритм назовем `DoubleBridging`. Сложность алгоритмов `Bridging` и `DoubleBridging` является полиномиальной, она зависит от метода определения расстояний между компонентами связности. В случае, если расстояния заданы ее можно оценить как $O(|E(G)| \cdot \log |V(G)|)$.

Теорема 5. *Если в каждой компоненте связности G_k графа G степени вершин, инцидентных разделяющим граням графа G, четны, то маршрут с минимальной длиной дополнительных построений реализуется алгоритмом `DoubleBridging`.*

В третьей главе «Построение маршрутов, удовлетворяющих комбинации локальных и глобальных ограничений» показана возможность распознавания системы переходов, которая позволяет решить задачу построения совместимого пути за линейное время. **Алгоритм P_G-СОВМЕСТИМАЯ ЭЙЛЕРОВА ЦЕПЬ**

Входные данные:

- эйлеров граф $G = (V, E)$, заданный списком смежности для каждой вершины;
- система переходов $P_G(v)\ \forall v \in V(G)$: в списке смежности вершины, относящиеся к одному элементу разбиения, имеют одинаковые пометки.

Выходные данные:

- допустимый эйлеров цикл G_{k+1}.

Шаг 1. Положить $k = 0$, $G_k = G$.

Шаг 2. Найти вершину v, у которой $d_{G_k}(v) > 2$.

Шаг 3. Найти элемент разбиения, который содержит максимальное число ребер:

- просмотреть список смежности текущей вершины v;
- посчитать число вхождений в этот список каждого элемента разбиения;
- выбрать тот элемент, который встречается чаще: получим класс $C_1 \in P_{G_k}(v) : |C_1| = \{\max |C| \,|C \in P_{G_k}(v)\}$.

Шаг 4. Выбрать ребра $e_1(v) \in C_1$ и $e_2(v) \in E_{G_k}(v) - C_1$. По возможности выбрать ребра e_1 и e_2, инцидентные вершинам, степень которых больше двух. Если множество $E_{G_k}(v) - C_1 = \emptyset$, останов: P_G-совместимой эйлеровой цепи не существует. В противном случае перейти на шаг 5.

Шаг 5. Построить граф G_{k+1}, отщепив от вершины v вершину $\widehat{v}$, которой инцидентны только ребра e_1 и e_2. Остальные ребра оставить инцидентными вершине v. Так как новая вершина имеет степень 2, то она не рассматривается на последующих шагах работы алгоритма.

Шаг 6. Выбрать класс $C_2 \in P_{G_k}(v)$, которому принадлежит ребро $e_2(v)$. Исключить из системы разбиения вершины v классы C_1 и C_2. Для этого нужно найти $P^-_{G_k}(v) := P_{G_k}(v) - \{C_1, C_2\}$.

Для дальнейшей модификации системы разбиений выполнить следующие действия.

Шаг 6.1. Системы разбиения, в которых отсутствует вершина v, перейдут в модифицированную систему полностью без изменений.

Шаг 6.2. Если системы C_1 и C_2 состояли из одного ребра: $|C_1| = |C_2| = 1$, то $P'_{G_{k+1}}(v) = P^-_{G_k}(v)$.

Шаг 6.3. Если $|C_1| > |C_2| = 1$, то $P'_{G_{k+1}}(v) = P^-_{G_k}(v) \cup \{C_1 - \{e_1(v)\}\}$.

Шаг 6.4. Если $|C_2| > 1$, то $P'_{G_{k+1}}(v) = P^-_{G_k}(v) \cup \{C_1 - \{e_1(v)\}, C_2 - \{e_2(v)\}\}$.

Шаг 6.5. Построить

$$P_{G_{k+1}} = \bigcup_{x \in V(G_{1,2})} P'_{G_{k+1}}(x).$$

Шаг 7. Определить значение $\sigma(G_{k+1}) = 2(|E(G_{k+1})| - |V(G_{k+1})|)$. Заметим, что количество ребер графа остается неизменным, а количество вершин на каждой итерации увеличивается на единицу.

Шаг 8. Если $\sigma(G_{k+1}) > 0$, положить $k = k+1$, перейти на шаг 2, для графа G_{k+1}. В противном случае перейти на шаг 9.

Шаг 9. Выбрать любую вершину v и пометить все вершины достижимые из данной. Если остались непомеченные вершины перейти на шаг 10, иначе останов – построенный граф G_{k+1} является эйлеровой цепью, не содержащей запрещенных переходов.

Шаг 10. Из списка помеченных и не помеченных вершин графа G_{k+1} найти вершины v_1 и v_2, отщепленные от одной вершины v графа G_0 и объединить их в вершину $v_{1,2}$. Получим модифицированный граф $\hat{G}_{k+1}$, положим $k = k+1$.

Шаг 11. Выбрать ребра $e_1(v_{1,2}) \in C_1$ и $e_2(v_{1,2}) \in E_{G_k}(v_{1,2}) - C_1$, так чтобы $\{e_1, e_2\} \neq E(v_1)$ и $\{e_1, e_2\} \neq E(v_2)$. Если множество $E_{G_k}(v_{1,2}) -$

$C_1 = \emptyset$, останов: P_G-совместимой эйлеровой цепи не существует. В противном случае построить граф G_{k+1}, отщепив от вершины $v_{1,2}$ вершину $\hat{v}_{1,2}$, которой инцидентны только ребра e_1 и e_2. Остальные ребра оставить инцидентными вершине $v_{1,2}$ и перейти на шаг 9.

Теорема 6. *Алгоритм P_G-СОВМЕСТИМАЯ ЭЙЛЕРОВА ЦЕПЬ корректно решает задачу построения $P(G)$-совместимой эйлеровой цепи.*

Покрытие графа G совместимыми цепями также возможно за время $O(|V(G)| \cdot |E(G)|)$ с помощью алгоритма `ПОКРЫТИЕ` T_G`-СОВМЕСТИМЫМИ ЦЕПЯМИ`.

Алгоритм ПОКРЫТИЕ T_G-СОВМЕСТИМЫМИ ЦЕПЯМИ

Входные данные:

- граф $G = (V, E)$,
- графы переходов $T_G(v)$ $\forall v \in V(G)$.

Выходные данные:

- набор цепей T^i, $i = 1, 2, \ldots, k$, покрывающих граф G, где $m = 2k$ – число вершин нечетной степени.

Шаг 1. Пусть $U = \{v \in V(G) \ : \ T_G(v) - \text{паросочетание}\}$. Сделать редукцию графа G до графа G':

$$V(G') = V(G) \backslash U,$$

$$E(G') = \left(E(G) \backslash \bigcup_{v \in U} E_G(v) \right) \bigcup \left\{ \bigcup_{v \in U} \{\{v_i v_j\} : \ \{v_i v, \ v v_j\} \in T_G(v)\} \right\},$$

графы $T_G(v)$ редуцировать до графов $T_{G'}(v)$ заменой всех вхождений вершин $u \in U$: $vu, wu \in E_{T_G}(u)$ вершиной w.

Шаг 2. Достроить граф G' до G^* введением дополнительной вершины v^*, смежной всем вершинам нечетной степени графа G'. Систему переходов $T_{G'}(v)$ модифицировать до системы переходов T_{G^*} введением для всех $v \in V'(G) \ : \ \deg(v) \equiv 1 \pmod 2$ в граф переходов $T_{G^*}(v)$ вершины vv^*, смежной всем вершинам в графе $T_{G^*}(v)$.

Шаг 3. Для всех таких вершин $v \in V(G)$, что $\exists p \in P(v)$: $|p| > \deg(v)/2$, ввести $2|p| - \deg(v)$ дополнительных ребер $(vv^*)_i$, $i = 1, 2, ..., 2\,|p| - \deg(v)$ в граф G^*. Модифицировать граф переходов $T_{G^*}(v)$ введением вершин $(vv^*)_i$, смежных всем вершинам исходного графа $T_{G^*}(v)$ и только им.

Шаг 4. Найти в G^* T_{G^*}-совместимый эйлеров цикл T^*.

Шаг 5. Построить покрытие T' графа G' цепями, удалив из T^* ребра (vv^*).

Шаг 6. Модифицировать маршруты из T' до маршрутов из T добавлением вершин $u \in U$, удаленных на шаге 1.

Шаг 7. Останов.

Теорема 7. *Алгоритм* ***ПОКРЫТИЕ*** T_G***-СОВМЕСТИМЫМИ ЦЕПЯМИ*** *корректно решает задачу минимального по мощности покрытия графа T_G-совместимыми цепями. Его сложность не превосходит величины $O(|E(G)| \cdot |V(G)|)$.*

Приведены примеры использования построенных алгоритмов.

При технологической подготовке процесса раскроя возникают различные ограничения на траекторию движения режущего инструмента. Так, выше была рассмотрена задача построения маршрута, при котором отрезанная часть листа не требовала дополнительных разрезаний. Однако на практике требуется выполнение дополнительного ограничения на отсутствие пересечения резов. Для решения такой задачи используется цепь с наложенным на нее локальным ограничением. В каждой вершине графа задан циклический порядок обхода ребер, и продолжение обхода по цепи осуществляется только в соответствии с этим циклическим порядком. В общем случае задача поиска такой цепи в графе относится к классу $\mathcal{NP}$-полных задач, однако для некоторых частных случаев существуют эффективные алгоритмы ее решения.

Рассмотрим эйлерову цепь

$$T = v_0, k_1, v_1, \ldots, k_n, v_n,\ v_n = v_0$$

в графе $G = (V, E)$. Предположим, что в каждой вершине $v \in V$ задан циклический порядок $O^{\pm}(v)$, определяющий систему переходов $A_G(v) \subseteq O^{\pm}(v)$ в этой вершине. В случае, когда $\forall v \in V(G)$ $A_G(v) = O^{\pm}(v)$, систему переходов $A_G(v)$ будем называть **полной системой переходов**.

Определение 6. *A_G-совместимую цепь T будем называть* ***A-цепью****. Таким образом, последовательные ребра в цепи T (инцидентные вершине v) являются соседями в циклическом порядке $O^{\pm}(v)$.*[5]

[5] *Фляйшнер Г.* Эйлеровы графы и смежные вопросы. М.: Мир, 2002. 335 с., ил.

Определение 7. *Будем говорить, что цепь является AOE-цепью, если она одновременно является OE-цепью и A-цепью.*

Теорема 8. *Если в плоском графе G существует A-цепь, то существует и AOE-цепь.*

Более того, показано, что OE-цепь существует и для произвольной системы переходов без самопересечений. Показано, что число OE-цепей, соответствующих фиксированной системе переходов равно удвоенному числу вершин графа, смежных внешней грани.

Теорема 9. *В плоском связном 4-регулярном графе G существует AOE-цепь.*

Рассмотрен алгоритм `AOE-TRAIL` (см. алг. 5) построения AOE-цепи в плоском связном 4-регулярном графе.

Определение 8. *Суграф G_k графа G, для которого $E(G_k) = \{e \in E(G) : \mathrm{rank(e)} \geq \mathrm{k}\}$ назовем **суграфом ранга** k.*

Предварительно производится «правильное» расщепление всех точек сочленения суграфов G_k, чтобы в результате расщепления получить граф, у которого любой суграф G_k не содержит точек сочленения. Последовательность расщепления вершин не имеет значения, т.к. это локальная операция. Под «правильным» будем понимать переход между дугами, соответствующими циклическому порядку и инцидентными различным парам граней (см. рис. 3.а)). Результат расщепления приведен на рис. 3.б). При поиске точек сочленения используются следующие свойства 4-регулярных графов.

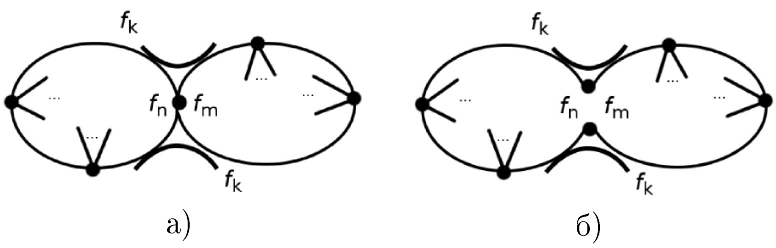


Рис. 3. Расщепление точек сочленения ранга k. а) Верная система переходов для расщепления точки сочленения суграфа G_k. б) Расщепление в соответствии с системой переходов в точке сочленения суграфа G_k

Algorithm 5 Алгоритм AOE-`TRAIL`

Require: плоский связный 4-регулярный граф $G = (V, E)$ без точек сочленения, представленный функциями v_k, l_k, r_k, $k = 1, 2$;   начальная вершина $v \in V(f_0)$.

Ensure: $ATrail$ – выходной поток, содержащий построенную алгоритмом AOE-цепь.

Initiate(G, v_0); ▷ Инициализация
Ranking(G); ▷ Ранжирование
CUT_POINT_SPLITTING (G); ▷ Удаление точек сочленения в суграфах всех рангов
▷ Построение
$e = \arg\max_{e \in E(v)} \operatorname{rank}(e)$ ▷ Выбрать ребро максимального ранга, инцидентное вершине v
repeat
  if $v \neq v_1(e)$ **then**
    REPLACE(e)
  end if ▷ При необходимости скорректировать нумерацию функций для ребра e
  Print(v, e) ▷ Вывести ребро e в результирующую последовательность
  mark(e) := **false**; counter:=counter+1; $v := v_2(e)$ ▷ Пометить ребро как пройденное
  if (rank($r_2(e)$) $\geq$ rank($l_2(e)$)) **then** ▷ Выбрать следующим ребро максимального ранга
    if mark($r_2(e)$) **then** ▷ Проверить, пройдено ли добавляемое в цепь ребро
      $e := r_2(e)$ ▷ Для пройденных ребер значение в массиве $mark$ — ложь
    else
      $e := l_2(e)$
    end if
  else
    if (mark($l_2(e)$) **then** ▷ Выбрать непройденное ребро
      $e := l_2(e)$
    else
      $e := r_2(e)$
    end if
  end if
until (counter $> |E(G)|$) ▷ Завершить цикл, когда просмотрены все ребра
Конец Алгоритма

Предложение 1. *Вершина, инцидентная четырем ребрам, смежным внешней грани, является точкой сочленения.*

Предложение 2. *Внешняя грань суграфа G_k является объединением всех граней ранга k в графе G.*

Из них следует результативность приведенного в диссертацинной работе алгоритма `CUT-POINT-SPLITTING` (см. алг. 6), вычислительная сложность которого не превосходит $O(|E(G)| \log |V(G)|)$.

Теорема 10. *Алгоритм AOE-`TRAIL` строит AOE-цепь в плоском связном 4-регулярном графе G, любой суграф G_k, $k = 1, 2, \ldots$ которого не содержит точек сочленения. Алгоритм находит решение за время $O(|E(G)| \cdot \log |V(G)|)$.*

Algorithm 6 CUT-POINT-SPLITTING

Require: плоский связный 4-регулярный граф $G = (V, E)$, представленный для всех $e \in E(G)$ функциями v_s, l_s, r_s , $s = 1, 2$.

Ensure: гомеоморфный образ графа $G = (V, E)$, представленный для всех $e \in E(G)$ функциями v_s, l_s, r_s , $s = 1, 2$, в котором любой суграф G_k не имеет точек сочленения.

Промежуточные данные: $\forall v \in V(G)$: $\mathrm{point}(v)$ — массив указателей на одно из ребер, инцидентных вершине v, $\mathrm{rank}(v)$ — массив рангов вершин, $\mathrm{count}(v)$ — счетчик инцидентных ребер одного ранга для каждой вершины;
$\forall f \in F(G)$: массив $\mathrm{rank}(f)$.

Initiate():
for all $v \in V(G)$ **do** ▷ Обнулить счетчик инцидентных вершине ребер одного ранга
    $\mathrm{point}(v) := 0$; $\mathrm{count}(v) := 0$
end for
Ranking(G) ▷ Определение ранга всех вершин, ребер и граней графа
Finding(): ▷ Нахождение точек сочленения
for all $e \in E(G)$ **do**
    $\mathrm{point}(v_1(e)) := e$
    $\mathrm{point}(v_2(e)) := e$
end for
for all $v \in V(G)$ **do** ▷ Просмотреть последовательно все вершины
    $e := \mathrm{point}(v)$
    $k := \mathrm{rank}(v)$ ▷ Сохранить значение ранга k инцидентного ребра e
    if $v = v_1(e)$ **then** ▷ Определить ориентацию ребра e
        $s := 1$
    else
        $s := 2$
    end if
    $e := l_s(e)$ ▷ Подсчет количества инцидентных вершине v ребер ранга k
    for $i = 1$ **up to** 4 **do**
        if $\mathrm{rank}(e) = k$ **then**
            $\mathrm{count}(v) := \mathrm{count}(v) + 1$
            if $i < 4$ **then**
                $e := l_s(e)$
            end if
        end if
    end for
    if $\mathrm{count}(v) = 4$ **then** ▷ Если найдена точка сочленения, расщепить вершину
        if $(f_s(e) = f_s(l_s(e))$ **and** $f_{3-s}(e) = f_{3-s}(l_s(e)))$ **or** $(f_s(e) = f_{3-s}(l_s(e))$ **and** $f_{3-s}(e) = f_s(l_s(e)))$ **then**
            $e^* := l_s(e)$, $l_s(e) := r_s(e)$, $r_s(r_s(e)) := e$,
            $r_s(e^*) := l_s(e^*)$, $l_s(l_s(e^*)) := e^*$
        else
            $e^* := r_s(e)$, $r_s(e) := l_s(e)$, $l_s(l_s(e) := e$,
            $l_s(e^*) := r_s(e^*)$, $r_s(r_s(e^*)) := e^*$
        end if
    end if
end for

Algorithm 7 `NOE-CHAIN (G)`

Require: плоский эйлеров граф G, заданный функциями $v_k(e)$, $l_k(e)$, $r_k(e)$, $f_k(e)$, $k = 1, 2$ и $\mathrm{rank}(e)$;

Ensure: NOE-цепь в графе G;

1: $\widehat{G} = \mathsf{NonIntersecting}(G)$; ▷ Расщепить все вершины степени выше 4
2: $\widetilde{G} = \mathsf{CutPointSplitting}(\widehat{G})$; ▷ Расщепить все точки сочленения всех рангов
3: C^*=$\mathsf{AOE\_TRAIL}(\widetilde{G})$; ▷ Построить AOE-цепь в графе $\tilde{G}$
4: C=$\mathsf{Absorb}(C^*)$; ▷ Стянуть все расщепленные вершины

Алгоритм AOE-`TRAIL` используется в алгоритме 7 `NOE-CHAIN` для построения самонепересекающейся OE-цепи в плоском связном графе. Его выполнение состоит в сведении исходного графа к плоскому связному 4-регулярному графу за счет расщепления вершин степени выше 4.

Для построения эйлерова NOE-цикла в плоском эйлеровом графе, для которого не задано фиксированной системы переходов, можно поступить следующим образом.

На множестве вершин графа $V(G)$ определим булеву функцию

$$\mathrm{Checked}(v) = \begin{cases} \texttt{true}, & \text{если вершина просмотрена;} \\ \texttt{false}, & \text{в противном случае.} \end{cases}$$

При выполнении инициализации все вершины объявить непросмотренными. Процедура `Non-intersecting` (G) (алгоритм 8) расщепляет в графе G все вершины $v \in V(G)$, степени $2k - 1$ или k ($k \geq 3$) на k искусственных вершин степени 4 и вводит k искусственных ребер, инцидентных полученным после расщепления вершинам и образующим цикл.

В теле функции используется процедура `Handle` (e, $v_k(e)$, k), которая обрабатывает каждую непросмотренную вершину графа G. Обработка заключается в расщеплении вершины $v_k(e)$ в соответствии с рис.4.

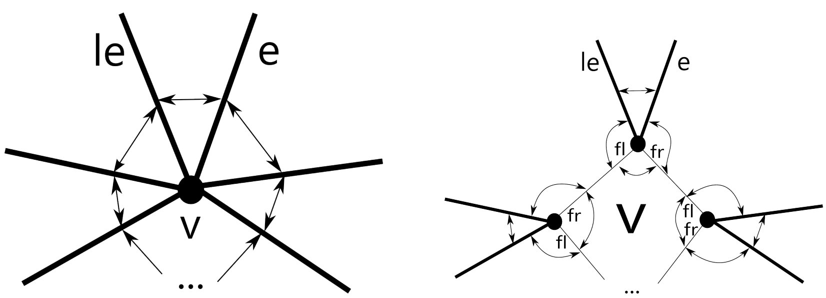


Рис. 4. Расщепление вершины (жирными линиями показаны ребра графа G, тонкими линиями – дополнительные (фиктивные) ребра) и модификация указателей в соответствии с расщеплением

Algorithm 8 Функция `Non-intersecting (G)`

Require: плоский эйлеров граф G, заданный функциями $v_k(e)$, $l_k(e)$, $r_k(e)$, $f_k(e)$, $k = 1, 2$ и $\mathrm{rank}(e)$;

Ensure: плоский связный 4-регулярный граф G^*, определяемый аналогичным образом;

```
for all v ∈ V(G) do                          ▷ Инициализация функции Checked(v)
    Checked(v) := false;
end for
for all (e ∈ E(G) ) do                       ▷ Поиск вершин степени больше 4 и их расщепление
    k := 1;                                  ▷ Просмотреть вершину с индексом 1, затем – 2
    while (k ≤ 2) do
        if (! Checked(v_k(e))) then          ▷ Обработать только не обработанную ранее вершину
            if (k = 2) then                  ▷ Скорректировать индексы
                REPLACE(e);                  ▷ обрабатываются вершины v_1(e)
            end if
            Handle ( e);                     ▷ Вызвать функцию для обработки вершины v_1(e)
            Checked(v_1(e)) := true;         ▷ Пометить вершину как просмотренную
        end if
        k := k + 1;
    end while
end for
    Конец Функции
```

Введенные процедурой `Handle` (алгоритм 9) $k/2$ искусственных вершин и k искусственных ребер, инцидентных этим вершинам, образуют цикл. В результате обработки всех вершин графа G получим модифицированный граф G^*, являющийся плоским связным 4-регулярным графом. Для G^* можно применить алгоритм `AOE-TRAIL()`, который построит в нем AOE-цепь T^*. Отписанная процедура реализована алгоритмом `Non-intersecting` (см. алгоритм 8). Если затем в T^* все искусственные ребра и инцидентные им вершины, полученные при расщеплении вершины v, заменить на v, то получим NOE-цепь T в исходном графе G.

Отметим, что данный алгоритм строит NOE-цепь в плоском эйлеровом графе. В случае плоского неэйлерова (в общем случае несвязного) графа G необходимо расщепить все вершины степени выше 4 в соответствии с алгоритмом 9. В результате получим граф, степени вершин которого равны 3 или 4. Для этого графа применим алгоритм построения AOE-покрытия. В цепях полученного покрытия удалим все искусственные ребра и стянем все расщепленные вершины. В результате получим NOE-покрытие.

Задача пересчета OE-цепей для плоского графа имеет большой практический интерес, т.к. ее решение позволяет, например, определить возможность раскроя по допустимой траектории из различных началь-

Algorithm 9 Процедура Handle (e)

```
procedure HANDLE(e)
    v := v1(e);                          ▷ Расщепляемая вершина
    e_first := e;                        ▷ Сохранить первое рассматриваемое ребро
    d := 0;                              ▷ Инициализация счетчика для степени вершины d
    F := FaceNum() + 1;                  ▷ Определить номер для новой грани
    repeat                               ▷ Проход 1: Определение степени вершины v
        le := l1(e);
        if (v1(le) ≠ v) then REPLACE(le);
        end if                           ▷ При необходимости поменять индексацию функций
        e := le; d := d + 1;             ▷ Учесть ребро при подсчете степени и перейти к следующему
    until (e = e_first);                 ▷ Повторять, пока не будут просмотрены все ребра, инцидентные v
    if (d > 4) then                      ▷ Если степень текущей вершины больше 4
        e := e_first;                    ▷ Начать с первого рассматриваемого ребра
        le := l_k(e);                    ▷ Определить номер его левого соседа
        e_next := l_k(le);               ▷ Сохранить ребро, для следующей итерации
        fl := new EDGE; fle := fl; e_first := e;   ▷ Ввести фиктивное ребро, смежное le
        repeat                           ▷ Расставить указатели для ребер
            e := e_next; le := l_k(e); fr := fl;
            f1(fl) := F; f2(fl) := f2(e);  ▷ Определить грани, смежные фиктивному ребру
            rank(fl) := facerank(f2(fl));  ▷ Определить «ранг» фиктивного ребра
                ▷ Функция facerank() вычисляет ранг грани в соответствии с определением
            fl := new EDGE; e_next := l_k(le);
        until (l_k(le) = e_first);
    end if
end procedure
```

ных точек. Рассмотрим плоский эйлеров граф G и OE-цепь T в этом графе. Пусть $X_T(G)$ – система переходов, соответствующая цепи T, тогда верно следующе утверждение.

Предложение 3. *Пусть $G(V, E)$ – плоский эйлеров граф без точек сочленения и T представляет OE-цепь в графе G, которой соответствует система переходов $X_T(G)$. Тогда число OE-цепей N для системы переходов $X_T(G)$ удовлетворяет неравенству $1 \leq N \leq 2 \cdot |V(f_0)|$, $V(f_0) = \{v|\ v \in f_0\}$, причем как верхняя, так и нижняя оценки достижимы.*

С практической точки зрения особый интерес представляют графы, для которых верхняя оценка достижима. Не всякий OE-цикл индуцирует систему переходов, для которой будет достигаться верхняя оценка. Отметим также, что для нахождения подходящей системы переходов, для которой достигается верхняя оценка, недостаточно знать только начальную вершину и начальное ребро.

Теорема 11. *Пусть плоский граф $G = (V, E)$ имеет K разделящих вершин $v_1, \dots v_K \in f_0$ и пусть в этом графе существует A-цепь T. Пусть $X_T(G)$ – система переходов, соответствующая T, а $V(f_0)$ – множество вершин, смежных внешней грани. Существует*

$$2 \cdot |V(f_0)| + \sum_{i=1}^{K} (\deg(v_i) - 2)$$

OE-циклов для $X_T(G)$.

В четвертой главе «Практическое применение разработанных алгоритмов маршрутизации» показано, каким образом разработанные алгоритмы могут быть применены на практике на примере решения задачи построения программы управления раскройным автоматом. Отмечено, что в отличие от гильотинного раскроя, негильотинный раскройный план не дает программу вырезания деталей. Построение программы управления раскройным автоматом для реализации заданного раскройного плана является самостоятельной задачей. Приведена классификация задач маршрутизации инструмента машин листовой резки, предложенная в работах Дж. Хоэфта и У. С. Палекара[6].

Показано, что технологии ECP и ICP за счет возможности совмещения границ вырезаемых деталей позволяют сократить расход материала, длину резки, и длину и количество холостых проходов. Проблемы уменьшения отходов материала и максимального совмещения фрагментов контуров вырезаемых деталей решается на этапе составления раскройного плана.

Отмечено, что применение технологий ECP и ICP в системе технологической подготовки процессов раскроя плоских деталей предполагает следующие этапы.

1. Составление раскройного плана, заключающееся в нахождении такого варианта размещения вырезаемых деталей на прямоугольном листе или ленте, при котором минимизируются отходы и максимизируется длина совмещенных элементов контуров вырезаемых деталей.

2. Абстрагирование раскройного плана до плоского графа. Для определения последовательности резки фрагментов раскоройного плана не

[6] *Hoeft J., Palekar U.S.* Heuristics for the plate-cutting travelling salesman problem // IIE Transactions, 1997, no.29(9), P. 719–731.

используется информация о форме детали, поэтому все кривые без самопересечений и соприкосновений на плоскости, представляющие форму деталей, интерпретируются в виде ребер графа, а все точки пересечений и соприкосновений представляются в виде вершин графа. Для анализа выполнения технологических ограничений необходимо введение дополнительных функций на множестве вершин, граней и ребер полученного графа.

3. Решение задачи построения оптимальных маршрутов с ограничениями, наложенными на порядок обхода ребер. Данные ограничения непосредственно вытекают из технологических ограничений, наложенных на порядок вырезания деталей.

4. Составление программы управления процессом раскроя на основе маршрута, найденного с помощью алгоритма решения абстрагированной задачи маршрутизации. Здесь выполняется обратная замена абстрактных ребер плоского графа системой команд раскройному автомату, обеспечивающей движение по кривым на плоскости, соответствующим форме вырезаемой детали.

Этапы построения раскройного плана и интерпретации найденного маршрута в терминах команд раскройному автомату являются общими для всех технологий и достаточно известны. Реализация второго и третьего этапов для технологий ECP и ICP возможна применением разработанных алгоритмов построения OE-покрытий.

Рассмотрена задача, возникающая в случае воникновения ограничений на расположение точек врезки. Очевидно, что число точек врезки определяется количеством покрывающих цепей. В соответствии с теоремой 2 количество точек врезки будет не менее $|V_{odd}|/2$.

Задача может быть формализована следующим образом.

Пусть грани $F_{in}(G) \subset F(G)$ допускают врезку. Пусть вершины $V_{in}(G) \subset V(G)$ нечетной степени инцидентны грани $F_{in}(G)$. Если построенный маршрут в графе является OE-маршрутом и все начальные вершины покрывающих цепей принадлежат $V_{in}(G)$, тогда этот маршрут можно использовать как основу для построения маршрута движения режущего инструмента при лазерной резке. Такие маршруты будем называть $PPOE$-маршрутами.

Определение 9. *Цепь $C = v_1e_1v_2e_2\dots v_k$ будем называть* ***PPOE-цепью****, если она является OE-цепью и начинается в вершине $v_1 \in V_{in}(G)$.*

Определение 10. ***PPOE-покрытием*** *графа G будем называть OE-покрытие графа G, состоящее из PPOE-цепей.*

Определение 11. *Минимальную по мощности упорядоченную последовательность реберно-непересекающихся PPOE-цепей в плоском графе G будем называть* ***эйлеровым PPOE-покрытием****.*

Задачу определения реализуемости раскройного плана можно сформулировать как определение существования эйлерова $PPOE$-покрытия для плоского графа, являющегося гомеоморфным образом соответствующего раскройного плана. В соответствии с имеющимися ограничениями можно сформулировать следующее необходимое условие существования $PPOE$-покрытия.

Предложение 4. *Для существования PPOE-покрытия в плоском связном графе G необходимо, чтобы мощность минимального $\{V_{in},\ V_{out}\}$-разреза была не больше $|V_{out}|$.*

Рассмотрен декомпозиционный алгоритм нахождения $PPOE$-маршрутов.

Алгоритм *PPOE-маршрутизация*

Вход: плоский граф $G(V, E)$, заданный функциями $v_k(e)$, $l_k(e)$, $r_k(e)$, $f_k(e)$, $k = 1, 2$ и $\mathrm{rank}(e)$; множества $V_{out}, V_{in} \subset V$.

Выход: PPOE-покрытие графа $G(V, E)$:

$$\tilde{C}_1,\ \tilde{C}_2,\ \dots \tilde{C}_{|V_{out}|}, C_{|V_{out}|+1},\ \dots,\ C_M.$$

Шаг 1. Построить сеть $N(V, A)$ (т.е. ориентированный граф), в которой каждому ребру $e = \{u, v\} \in E(G)$ соответствует пара дуг $(u, v)\,, (v, u) \in A(N)$ с пропускной способностью 1; вершины $v^+ \in V_{out}$, т.е. точки возможного окончания цепи, являются источниками потока единичной мощности, вершины $v^- \in V_{in}$, т.е. возможные точки врезки, являются стоками.

Шаг 2. В сети $N(V,A)$ построить циркуляцию $x: A \to \{0,1\}$, т.е. решение задачи

$$f(x) = \sum_{(u,v)\in A(N)} x(u,v) \to \min_x,$$

$$\begin{aligned}
&\sum_{v:\,(u,v)\in A(N)} x(u,v) - \sum_{v:\,(v,u)\in A(N)} x(v,u) = 1, && u \in V_{out},\\
&\sum_{v:\,(u,v)\in A(N)} x(u,v) - \sum_{v:\,(v,u)\in A(N)} x(v,u) \ge -1, && u \in V_{in},\\
&\sum_{v:\,(u,v)\in A(N)} x(u,v) - \sum_{v:\,(v,u)\in A(N)} x(v,u) = 0, && u \in V \backslash (V_{out} \cup V_{in}),
\end{aligned}$$

$$0 \le x(u,v) \le 1, \quad (u,v) \in A(N).$$

Циркуляция $x: A \to \{0,1\}$ может быть построена с помощью решения задачи о максимальном потоке минимальной стоимости в двухполюсной сети, которая получается добавлением в сеть $N(V,A)$ общего источника s, смежного всем источникам, а также общего стока, смежного всем возможным стокам. При этом допустимой циркуляции $x: A \to \{0,1\}$ будет соответствовать величина максимального потока, не превышающая мощность множества V_{out}.

Шаг 3. Если $f(x) < |V_{out}|$, то в соответствии с предложением 4 PPOE-покрытие не существует, перейти на шаг 10.

Шаг 4. Для каждой активной дуги $(u,v): \ x(u,v) = 1$ создать список, включающий эту дугу и только ее.

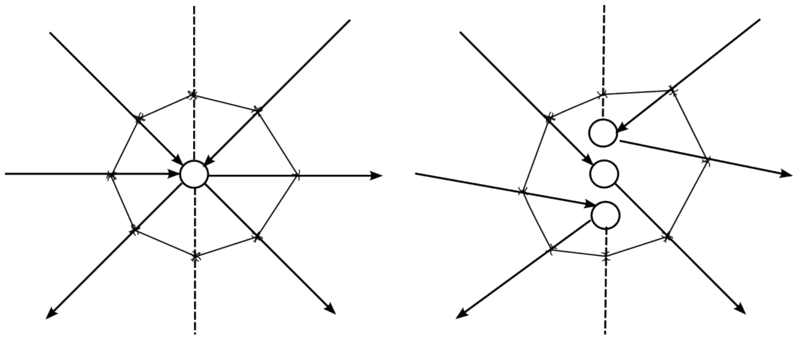
Шаг 5. Для каждой вершины $v \in V(G)$ пока есть возможность выполнять операции «правильного» расщепления вершин с «правильным» объединением списков активных дуг (т.е. с учетом циклического порядка на множестве дуг и их ориентации). Пример «правильных» расщепления и объединения приведены на рисунке. Результатом данного шага будет последовательность непересекающихся цепей C_1, C_2, $\ldots C_{|V_{out}|}$, содержащих все носители потока и только их.

Шаг 6. В построенных цепях C_1, C_2, $\ldots C_{|V_{out}|}$ изменить ориентацию дуг на противоположную.

Шаг 7. Продолжить каждую из цепей последовательности

$$C_1,\ C_2,\ \ldots C_{|V_{out}|}$$

до максимально возможной OE-цепи. Результатом выполнения данного шага будет начальная часть OE-покрытия $\tilde{C}_1$, $\tilde{C}_2$, $\ldots \tilde{C}_{|V_{out}|}$.



(а) (б)

Рис. 5. Пример: а) вершина и инцидентные ей ребра; б) «правильное» расщепление

Шаг 8. Построить суграф

$$\tilde{G} = G \setminus (\bigcup_{i=1}^{|V_{out}|} \tilde{C}_i),\ E(\tilde{G}) = \left(E(G) \setminus \left(\bigcup_{i=1}^{|V_{out}|} \tilde{C}_i\right)\right),$$

где все вершины $v \in V_{out}$, в которые запрещена врезка, будут вершинами четной степени.

Шаг 9. На множестве $V_{odd}(\tilde{G})$ найти кратчайшее паросочетание M. Для $\tilde{G}$ применить алгоритм M-`Cover` (см. алгоритм 3). Будут получены цепи $C_{|V_{out}|+1}, \ldots, C_{|V_{out}|+|M|}$.

Шаг 10. Останов.

Теорема 12. *Алгоритм $PPOE$-маршрутизация корректно решает задачу построения $PPOE$-покрытия в плоском графе $G(V, E)$ без мостов за время, не превосходящее $O(|V|^3)$.*

В заключении в краткой форме излагаются основные результаты проведенного диссертационного исследования.

Основные результаты работы

1. Введен класс маршрутов с упорядоченным охватыванием (OE-маршрутов) в плоских графах. Маршруты введенного класса представляют упорядоченную последовательность цепей, удовлетворяющую требованию отсутствия пересечения внутренних граней пройденной части маршрута с ребрами его непройденной части.

2. Показано, что на мощность покрытия существенное влияние оказывает наличие мостов в графе. При их отсутствии минимальное число

OE-цепей, покрывающих граф, совпадает с минимальным числом цепей, покрывающих данный граф (в случае существования вершин нечетной степени, инцидентных внешней грани). Если вершин нечетной степени, смежных внешней грани, нет, то мощность покрытия на единицу выше минимального числа цепей, покрывающих данный граф. В общем случае для мощности N эйлерова OE-покрытия графа G имеет место неравенство $k = \frac{|V_{odd}(G)|}{2} \leq N \leq |V_{odd}(G)| = 2k$. Верхняя и нижняя границы достижимы.

3. Разработаны полиномиальные алгоритмы построения OE-маршрута для разных случаев: плоский эйлеров граф, произвольный плоский связный граф (задача китайского почтальона и задача построения OE-покрытия), произвольный несвязный граф.

4. Для плоских графов без мостов разработаны алгоритм нахождения OE-маршрута с минимальным количеством покрывающих цепей, и алгоритм построения OE-маршрута с минимальной длиной переходов между концом текущей и началом следующей цепей. Все разработанные алгоритмы имеют полиномиальную сложность.

5. Введен класс AOE-цепей, в котором на цепь наложено локальное ограничение: смежные ребра цепи соответствуют системе переходов A-цепи. Разработан алгоритм `AOE-TRAIL`, который позволяет построить AOE-цепь для плоского связного 4-регулярного графа. Алгоритм находит решение за время $O(|E(G)| \cdot \log |V(G)|)$.

6. Введен класс NOE-маршрутов в плоских графах. Этот класс является расширением класса AOE и в него входят все OE-цепи, имеющие непересекающиеся переходы. Разработан алгоритм `Non-intersecting` построения NOE-цепи. Его выполнение состоит в сведении исходного плоского графа к плоскому связному 4-регулярному графу за счет расщепления вершин степени выше 4 и дальнейшего выполнения алгоритма `AOE-TRAIL`.

7. Определены оценки количества OE-цепей в эйлеровом графе для фиксированной системы переходов. Решение данной задачи может быть полезно при генерации допустимых вариантов маршрутизации.

Публикации автора по теме диссертации

Статьи, опубликованные в научных журналах и изданиях, которые включены в перечень ВАК

1. *Панюкова Т.А.* [7] Обходы с упорядоченным охватыванием в плоских графах // Дискретный анализ и исследование операций. Сер. 2. 2006. Т. 13, № 2. С. 31–43.
2. *Панюкова Т.А.* Последовательности цепей с упорядоченным охватыванием // Известия Академии наук. Теория и системы управления. 2007. № 1. С. 88–97. [8]
3. *Панюкова Т.А.* Маршруты с локальными ограничениями // Вестник Южно-Уральского государственного университета. Серия: Математическое моделирование и программирование. 2010. Вып. 5. № 16(192). С. 58–67.
4. *Панюкова Т.А.* Оптимальные эйлеровы покрытия с упорядоченным охватыванием для плоских графов // Дискретный анализ и исследование операций. 2011. Том 18, № 2. С. 64–74.
5. *Панюкова Т.А.* Оптимизация использования ресурсов при технологической подготовке процессов раскроя // Прикладная информатика. 2012. № 3(39). С. 20–32.
6. *Панюкова Т.А., Алферов И.О.* Маршруты с локальными ограничениями: алгоритмы и программная реализация // Прикладная информатика. 2013. № 1(43). С. 80–90.
7. *Панюкова Т.А., Савицкий Е.А.* The Software for Algorithms of Ordered Enclosing Covering Constructing for Plane Graphs // Вестник УГАТУ. 2013. Vol. 17, no. 6 (59). P. 39–44.
8. *Panyukova T.A.* Constructing of OE-postman Path for a Planar Graph // Вестник Южно-Уральского государственного университета. Серия: Математическое моделирование и программирование. 2014. Т. 7, № 4. С. 90–101.
9. *Макаровских Т.А., Савицкий Е.А.* Абстрагирование раскройного плана до плоского графа для эффективного решения задачи вырезания деталей// Вестник УГАТУ. 2015. Т.19, №3(69). С 190–196.

[7] Смена фамилии на основании Свидетельства о заключении брака от 11.10.2014 г. II-ИВ № 07829.

[8] *Panyukova T.* Chain Sequences with Ordered Enclosing // Journal of Computer and System Sciences International. 2007. Vol. 46, No. 1. P. 83–92.

10. *Макаровских Т.А.* О числе ОЕ-цепей для заданной системы переходов// Вестник Южно-Уральского государственного университета. Серия «Математика. Механика. Физика». 2016. Т.8. №1. С. 5–12.
11. *Макаровских Т.А., Панюков А.В., Савицкий Е.А.* Математические модели и алгоритмы маршрутизации для САПР технологической подготовки процессов раскроя // Автоматика и телемеханика, 2017, № 4, с. 123–140. [9]
12. *Макаровских Т.А.* Оценка мощности OE-покрытия плоского графа // Вестник УГАТУ. 2017. Т. 21, №2(76). С. 112–118.
13. *Макаровских Т.А.* Программное обеспечение для построения А-цепей с упорядоченным охватыванием в плоском связном 4-регулярном графе // Вестник Южно-Уральского государственного университета. Серия «Вычислительная математика и информатика». 2019. Т.8. №1. С. 36–53.
14. *Макаровских Т.А.* Построение самонепересекающихся ОЕ-маршрутов в плоском эйлеровом графе // Вестник Южно-Уральского государственного университета. Серия «Вычислительная математика и информатика». 2019. Т.8. №4.

Статьи, опубликованные в научных журналах и изданиях, индексируемых в SCOPUS

15. *Panyukova T.* Eulerian Cover with Ordered Enclosing for Flat Graphs// Electronic Notes in Discrete Mathematics. 2007. No. 28. P. 17–24.
16. *Panyukova T., Mukhacheva E.A.* Mathematical Models for Cutting Process Design // IFAC Proceedings Volumes. 13th IFAC Symposium on Information Control Problems in Manufacturing, INCOM'09. Moscow, 2009. P. 1085–1090.
17. *Makarovskikh T., Savitskiy E.* Algorithms for Constructing Resource-Saving Cutting Machines// Procedia Engineering. 2015. Vol. 129. P. 781–786.
18. *Makarovskikh T.A., Panyukov A.V., Savitsky E.A.* Mathematical Models and Routing Algorithms for CAM of Technological Support of Cutting Processes // ScienceDirect IFAC-PapersOnLine 49-12 (2016) 821–826.

[9] *Makarovskikh T.A., Panyukov A.V., Savitsky E.A.* Mathematical Models and Routing Algorithms for CAD Technological Preparation of Cutting Processes // Automation and Remote Control, 2017, Vol. 78, No. 4, pp. 868–882.

19. *Makarovskikh T.A., Panyukov A.V.* AOE-Trails Constructing for a Plane Connected 4-Regular Graph. // CEUR Workshop Proceedings. Vol 1623. Pp. 62–71. Online: htpp://ceur-ws.org/Vol-1623
20. *Makarovskikh T.* A-trails and their application to industrial process // 2nd International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM). 2016. P. 1–6, DOI: 10.1109/ICIEAM.2016. 7911672.
21. *Makarovskikh, T.A., Savitsky, E.A.* The OE-cover for a plane graph by chains with allowed starting vertices, 2017, CEUR Workshop Proceedings, Vol. 2064, pp. 103-111
22. *Makarovskikh, T.A., Panyukov, A.V.* The Cutter Trajectory Avoiding Intersections of Cuts, 2017, IFAC-PapersOnLine. 50(1), pp. 2284-2289.
23. *Makarovskikh, T., Panyukov, A.* Development of routing methods for cutting out details, 2018, CEUR Workshop Proceedings. Vol. 2098, pp. 249-263.
24. *Makarovskikh, T.A., Panyukov, A.V., Savitskiy, E.A.* Mathematical models and routing algorithms for economical cutting tool paths, 2018, International Journal of Production Research. Vol. 56(3), c. 1171-1188.

Монографии

25. *Макаровских Т.А.* Маршруты-покрытия специального вида в графах. Теоретические основы и применение в ресурсосберегающих технологиях. М.: ЛЕНАНД, 2018. – 216 с.

Свидетельства о регистрации программ для ЭВМ

26. *Панюкова Т.А.* Программа построения эйлеровых циклов с упорядоченным охватыванием (Euler Cycles Constructor): свид. о гос. рег. №2004610785; правообладатель Панюкова Т.А. – Заявка № 2004610193; заявл. 03.02.2004; зарегистр. 30.03.2004, реестр программ для ЭВМ.
27. *Панюкова Т.А.* Программа построения маршрутов с упорядоченным охватыванием (Ordered Routes Constructor): свид. о гос. рег. №2005612413; правообладатель Панюкова Т.А. – Заявка № 200562413; заявл 22.09.2005; зарегистр. 21. 11.2005, реестр программ для ЭВМ.
28. *Панюкова Т.А., Савицкий Е.А.* Программа построения оптимальных покрытий с упорядоченным охватыванием для многосвязных графов: свид. о гос. рег. №2011617777; правообладатель: ФГБОУ

ВПО «ЮУрГУ» (НИУ). – Заявка № 2011616000; заявл. 09.08.2011; зарегистр. 06.10.2011, реестр программ для ЭВМ.

29. *Панюкова Т.А., Алферов И.О.* Программа построения эйлеровой цепи с запрещенными переходами в графе: свид. о гос. рег. №2013661312, правообладатель: ФГБОУ ВПО «ЮУрГУ» (НИУ). – Заявка № 2013619032; заявл. 09.10.2013; зарегистр. 05.12.2013, реестр программ для ЭВМ.
30. *Макаровских Т.А.* Программа построения А-цепи с упорядоченным охватыванием в плоском 4-регулярном в графе: свид. о гос. рег. №2014663188, правообладатель: ФГБОУ ВПО «ЮУрГУ» (НИУ). – Заявка № 2014663188; заявл. 18.12.2014; зарегистр. 11.02.2015, реестр программ для ЭВМ.
31. *Макаровских Т.А., Панюков А.В., Савицкий Е.А.* Программа для оценки укладки деталей прямоугольной формы на плоскости по критерию плотности упаковки и ограничениям маршрута их вырезания: свид. о гос. рег. №2016662723, правообладатель: ФГБОУ ВПО «ЮУрГУ» (НИУ). – Заявка № 2016660049; заявл. 28.09.2016; зарегистр. 21.11.2016, реестр программ для ЭВМ.

Прочие публикации

32. *Панюкова Т.А., Панюков А.В.* Эйлеровы циклы с упорядоченным охватыванием // Проблемы теоретической кибернетики. Тезисы докладов XII Международной конференции (Нижний Новгород, 17–22 мая, 1999 г.). Под ред. Лупанова О.Б. М.: Изд-во механико-математического факультета МГУ, 1999. Ч. II. С. 148.
33. *Панюкова Т.А.* Построение эйлеровых циклов специального вида // Научные труды молодых исследователей программы «Шаг в будущее». М.: НТА «Актуальные проблемы фундаментальных наук», 1999. Т. 1. С. 108.
34. *Panioukova T.A., Panyukov A.V.* The Algorithm for Tracing of Flat Euler Cycles with Ordered Enclosing // Известия Челябинского научного центра УрО РАН. 2000. № 4(9). P. 18–22.
35. *Панюкова Т.А.* Построение эйлеровых циклов специального вида в планарном графе // Материалы VII Международного семинара «Дискретная математика и ее приложения»; Ч. II. Под ред. О. Б. Лупанова. М.: Изд-во центра прикладных исследований при механико-математическом факультете МГУ, 2001. С. 149.

36. *Панюкова Т.А.* Синтез программ управления процессом раскроя // Обозрение прикладной и промышленной математики. Под ред. Прохорова Ю.В. М.: Научное изд-во «ТВП», 2001. Т. 8, Вып. 2. С. 664.
37. *Панюкова Т.А.* К задаче об эйлеровых циклах с упорядоченным охватыванием/ Т.А. Панюкова//Тезисы Второго Международного конгресса студентов, молодых ученых и специалистов «Молодежь и наука – третье тысячелетие»/YSTM'02 (15–19 апреля 2002 г.). Часть 2. – М.: Издание региональной общественной организации научно-технической ассоциации «Актуальные проблемы фундаментальных наук», 2002. С. 33–34.
38. *Panyukova T., Panyukov A.* The Algorithm for Construction of Euler Cycles with Ordered Enclosing// Российская конференция «Дискретный анализ и исследование операций»: Материалы конференции (Новосибирск, 24–28 июня, 2002). Новосибирск: Издательство Института математики СО РАН, 2002. С. 147.
39. *Panioukova T.A., Panyukov A.V.* Algorithms for Construction of Ordered Enclosing Traces in Planar Eulerian Graphs // The International Workshop on Computer Science and Information Technologies' 2003, Proceedings of Workshop (Ufa, September 16–18, 2003). Ufa: USATU, 2003. Vol. 1. P. 134–138.
40. *Панюкова Т.А.* Алгоритмы построения обходов с упорядоченным охватыванием в плоских эйлеровых графах //Материалы Всероссийской конференции «Проблемы оптимизации и экономические приложения» (Омск, 1–5 июля 2003 г.). Омский филиал Института Математики СО РАН. Омск: Изд-во Наследие. Диалог Сибирь, 2003. С. 188.
41. *Панюкова Т.А.* Маршруты манипулятора, не содержащие запрещенных переходов // Труды XXXIV Уральского семинара «Механика и процессы управления». Т. 2. Миасс, 2004. С. 556–564.
42. *Панюкова Т.А.* Рекурсивный алгоритм построения обходов с упорядоченным охватыванием в плоских неэйлеровых графах // Материалы VIII Международного семинара «Дискретная математика и ее приложения» (Москва, 2–6 февраля, 2004 г.). М.: Изд-во механико-математического факультета МГУ, 2004. С. 444.

43. *Панюкова Т.А.* Эйлеровы циклы специального вида // Российская конференция «Дискретный анализ и исследование операций». Материалы конференции (Новосибирск, 28 июня–2 июля, 2004 г.). Новосибирск: Изд-во Ин-та математики, 2004. С. 147.
44. *Панюкова Т.А.* Построение маршрутов с упорядоченным охватыванием в плоских графах // Труды 36-й Региональной молодежной конференции «Проблемы теоретической и прикладной математики». Екатеринбург: УрО РАН, 2005. С. 61–66.
45. *Панюкова Т.А.* Маршруты с упорядоченным охватыванием // Труды VII международной конференции «Дискретные модели в теории управляющих систем» (Покровское, 4–6 марта, 2006 г.). М.: МАКС Пресс, 2006. С. 265–271.
46. *Панюкова Т.А.* Маршруты с локальными ограничениями // Труды 37-й Региональной молодежной конференции «Проблемы теоретической и прикладной математики». Екатеринбург: УрО РАН, 2006. С. 66–70.
47. *Panyukova T.* Eulerian Cover with Ordered Enclosing for Flat Graphs// Abstracts of Sixth Czech-Slovak International Symposium on Combinatorics, Graph Theory, Algorithms and Applications. Prague, Charles University, 2006. P. 103–104.
48. *Panyukova T.* Special Routes in Flat Graphs// Short abstracts of the international meeting "Euler and Modern Combinatorics" (St. Petersburg, June 1–7, 2007), 2007. P. 34–35.
49. *Панюкова Т.А.* Маршруты с локальными ограничениями // Информационно-математические технологии в экономике, технике и образовании: сборник тезисов Международной научной конференции. Екатеринбург: УГТУ-УПИ, 2007. С. 303–305.
50. *Panyukova T. Panyukova T.* The special routes in plane graphs // PAMM. Special Issue: Sixth International Congress on Industrial Applied Mathematics (ICIAM07) and GAMM Annual Meeting, Zurich, 2007, Published Online: 12 Dec 2008. Vol. 7. Issue 1. P. 2070015–2070016. DOI: 10.1002/pamm.200701066.
51. *Панюкова Т.А., Мухачева Э.А.* Проблема рационального использования промышленных материалов: оптимизация обратного хода рас-

кроя // Обратные задачи в приложениях. Сборник статей научно-практической конференции. Бирск: БирГСПА, 2008. С. 270–276.

52. *Панюкова Т.А.* Программное обеспечение для построения последовательностей цепей с упорядоченным охватыванием // Проблемы теоретической и прикладной математики: труды 39-й Региональной молодежной конференции. Екатеринбург: УрО РАН, 2008. С. 393–398.

53. *Панюкова Т.А., Мирасов В.Ф.* Построение совместимых цепей в графах // Проблемы теоретической и прикладной математики: тр. 39-й Регион. молодеж. конф. Екатеринбург, 2008. С. 38–43.

54. *Панюкова Т.А.* Построение маршрута для оптимального хода режущего инструмента // Информационно-математические технологии в экономике, технике и образовании. Тезисы докладов 3-й Международной научной конференции. Екатеринбург: УГТУ-УПИ, 2008. Ч. 2. С. 12–13.

55. *Panyukova T.* Routing problems for cutting processes // The International Workshop on Computer Science and Information Technologies' 2008. Proceedings of Workshop (Turkey, Antalya, September 15–17, 2008). Ufa: Ufa State Technical University, 2008. Vol. 2. P. 100–106.

56. *Панюкова Т.А.* Построение эйлерова покрытия с упорядоченным охватыванием для плоского графа с прямоугольными гранями // X Белорусская математическая конференция: Тез. докл. междунар. науч. конф. (Минск, 3–7 ноября 2008 г.). Мн.: Институт математики НАН Беларуси, 2008. Ч. 5. С. 98.

57. *Панюкова Т.А., Савицкий Е.А.* Композиция интерфейсов при технологической подготовке процессов раскроя // Труды 40-й Региональной молодежной конференции «Проблемы теоретической и прикладной математики». Екатеринбург: УрО РАН, 2009. С. 367–372.

58. *Панюкова Т.А.* Некоторые критерии оценки раскройных планов // IV Всероссийская конференция «Проблемы оптимизации и экономические приложения». Материалы конференции (Омск, 29 июня–4 июля, 2009). Омский филиал Института математики им. С.Л. Соболева СО РАН. Омск: Полиграфический центр КАН, 2009. С. 238.

59. *Панюкова Т.А., Панюков А.В.* Decreasing of Length for Routes with Ordered Enclosing // Дискретная математика, алгебра и их прило-

жения. Тез. Докл. Междунар. науч. конф. (Минск, 19–22 октября, 2009 г.). Мн.: Институт математики НАН Беларуси, 2009. С. 155–157.
60. *Панюкова Т.А., Савицкий Е.А.* О некоторых критериях оценки покрытий с упорядоченным охватыванием // Материалы X Международного семинара «Дискретная математика и ее приложения» (1–6 февраля, 2010 г.). М.: Изд-во механико-математического факультета МГУ, 2010. С. 444.
61. *Панюкова Т.А.* Покрытия с упорядоченным охватыванием с минимальной длиной дополнительных построений // Российская конференция «Дискретная оптимизация и исследование операций»: Материалы конференции (Алтай, 27 июня – 3 июля, 2010). Новосибирск: Изд-во Ин-та математики. 2010. С. 98.
62. *Panyukova T.* Ordered Enclosing Covers with Minimal Length of Additional Edges// Abstracts of 8th French Combinatorial Conference (Orsay, France, June,28–July,2, 2010). 2010. Abstract № 18.
63. *Panyukova T., Savitskiy E.* Optimization of Resources Usage for Technological Support of Cutting Processes // Proceedings of the Workshop on Computer Science and Information Technologies (Russia, Moscow–St. Petersburg, September 13–19, 2010). Ufa State Technical University, 2010. Vol. 1. P. 66–70.
64. *Панюкова Т.А., Алферов И.О., Сахаров И.А.* Об алгоритме построения совместимых путей в графе // Информационный бюллетень №12 Ассоциации математического программирования. XIV Всероссийская конференция «Математическое программирование и приложения». Екатеринбург, 2011. С. 120–121.
65. *Панюкова Т.А., Савицкий Е.А.* Допустимые эйлеровы покрытия с упорядоченным охватыванием для многосвязного графа // Статистика. Моделирование. Оптимизация: сб. тр. Всероссийской конференции (Челябинск, 28 ноября – 3 декабря 2011 г.). Челябинск: Издательский центр ЮУрГУ, 2011. С. 154–158.
66. *Panyukova T.* The Covering of Graphs by Trails with Local Restrictions// Proceedings oh the 13-th International Workshop on Computer Science and Information Technologies. Уфа: Издательство УГАТУ, 2011. Т. 1. С. 202–207.

67. *Панюкова Т.А.* Эйлерово покрытие с упорядоченным охватыванием для многосвязного плоского графа // Материалы V Всероссийской конференции (Омск, 2–6 июля, 2012 г.). 2012. С. 154.
68. *Панюкова Т.А.* Алгоритм построения оптимального эйлерова покрытия для многосвязного графа // Современные информационные технологии и ИТ-образование. Сборник избранных трудов VII Международной научно-практической конференции. Под ред. проф. В.А. Сухомлина. М.: ИНТУИТ.РУ, 2012. – С. 706–713.
69. *Панюкова Т.А., Алферов И.О.* Техника программной реализации алгоритма построения допустимой эйлеровой цепи // Информационные технологии и системы: материалы Первой междунар.конф. (Банное, Россия, 28.02–04.03.2012) / отв.ред. В.А. Мельников. Челябинск: Изд-во Челяб.гос.ун-та, 2012. С. 32–34.
70. *Панюкова Т.А.* Эйлерово покрытие с упорядоченным охватыванием для многосвязного графа // Математическое моделирование, оптимизация и информационные технологии: сборник научных трудов 3-й Международной конференции. Кишинев, 2012. С. 429–437.
71. *Панюкова Т.А.* Об оптимальном эйлеровом покрытии плоских графов // Информационные технологии и системы: тр. Второй междунар. науч. конф. (Банное, Россия, 27 февр.–3 марта 2013 г.): науч. электр. изд. / отв. ред. А. В. Мельников. Челябинск: Изд-во Челяб.гос.ун-та, 2013. С. 52–54.
72. *Панюкова Т.А., Савицкий Е.А.* Программное обеспечение для построения покрытия с упорядоченным охватыванием для многосвязных плоских графов // Вестник Южно-Уральского государственного университета: Серия "Вычислительная математика и информатика". 2013. Т. 2, № 2. С. 111–117.
73. *Panyukova T.* Covering with ordered enclosing for a multiconnected graph // Abstracts of the Seventh Czech-Slovak International Symposium on Graph Theory, Combinatorics, Algorithms and Applications (Kosice, Slovakia, 7–13 July, 2013), 2013. P. 65.
74. *Panyukova T.* On Algorithms for Eulerian Trails Constructing // Proceedings of the Workshop on Computer Science and Information Technologies (Vienna-Budapest-Bratislava, September 15–21, 2013). Ufa: USATU, 2013. Vol. 1. P. 176–181.

75. *Панюкова Т.А.* Построение эйлеровых циклов с упорядоченным охватыванием как математическая модель решения задачи раскроя // Современные информационные технологии и ИТ-образование. Сборник избранных трудов VIII Международной научно-практической конференции. Под ред. проф. В.А. Сухомлина. М.: ИНТУИТ.РУ, 2013. С. 706–713.
76. *Panyukova T., Alferov I.* The Software for Constructing Trails with Local Restrictions in Graphs // Open Journal of Discrete Mathematics. 2013. No. 3, Vol. 2. P. 86–92. DOI: 10.4236/ojdm.2013.32017.
77. *Панюкова Т.А.* Алгоритм покрытия плоского графа последовательностью цепей с упорядоченным охватыванием // Материалы международной конференции «Дискретная оптимизация и исследование операций». Новосибирск: Изд-во ин-та математики, 2013. С. 110.
78. *Панюкова Т.А.* О построении маршрута движения режущего инструмента при условии вырезания только соседних деталей // Информационные технологии и системы: тр. Третьей междунар. науч. конф. (Банное, Россия, 26 февр.–2 марта 2014 г.): науч. электр. изд. Отв. ред. Ю.С. Попков, А.В. Мельников. Челябинск: Изд-во Челяб. гос. ун-та, 2014. С. 41–42.
79. *Панюкова Т.А., Савицкий Е.А.* Алгоритм проверки маршрута реза в раскройном плане на сответствие условию упорядоченного охватывания // XII Всероссийское совещание по проблемам управления ВСПУ-2014. Институт проблем управления им. В.А. Трапезникова РАН. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2014. С. 9315–9318.
80. *Панюкова Т.А.* Комбинаторика и теория графов: Учебное пособие. М.: Книжный дом «ЛЕНАНД», 2014. 208 с.
81. *Панюкова, Т.А.* Модель движения режущего инструмента при условии вырезаний только соседних деталей // «Информационно-телекоммуникационные системы и технологии» (ИТСиТ-2014) Материалы Всероссийской научно-практической конференции. Кемерово, 2014. С. 411–412.
82. *Панюкова Т.А., Савицкий Е.А.* Использование двойственного граневого графа при построении маршрута режущего инструмента автомата раскроя// Информационные технологии интеллектуальной

поддержки принятия решений (Proceedings of the 2nd International Conference «Information Technologies for Intelligent Decision Making Support» and the Intended International Workshop «Robots and Robotic Systems»). 2014. С. 284-287.
83. *Panyukova T.* Covering with ordered enclosing for a disconnected graph// Proceedings of the Workshop on Computer Science and Information Technologies (Sheffield, England, September 16–22, 2014). Ufa: USATU, 2014. Vol. 1. P. 132–137.
84. *Панюкова, Т.А.* Модель движения режущего инструмента при условии вырезаний только соседних деталей // «Информационно-телекоммуникационные системы и технологии» (ИТСиТ-2014) Материалы Всероссийской научно-практической конференции. Кемерово, 2014. С. 411–412.
85. *Makarovskikh T.A.* On the number of starting points for a fixed cutting plan and fixed cutter trajectory // Proceedings of the 2-nd International Conference «Intelligent Technologies for Information Processing and Management». Ufa: USATU, 2014. Vol. 1. P. 239–244.
86. *Makarovskikh T.A.* The Algorithm for Constructing of Cutter Optimal Path // Journal of Computational and Engineering Mathematics. 2014. Vol. 1, №2. P. 52–61.
87. *Макаровских Т.А.* О построении эйлерова АОЕ-покрытия в плоском графе // Информационный бюллютень №13, XV Всероссийская конф. «Математическое программирование и приложения». 2015. С. 96–97.
88. *Макаровских Т.А.* Покрытие графа для прямоугольного раскройного плана АОЕ-цепями // Информационные технологии и системы: тр. Четвертой междунар. науч. конф., Банное, Россия, 25 февр. – 1 марта 2015 г. (ИТиС–2015) : науч. электр. изд. / отв. ред. Ю.С. Попков, А.В. Мельников. Челябинск : Изд-во Челяб. гос. ун-та, 2015. – С.17–18.
89. *Макаровских Т.А.* Определение траектории движения режущего инструмента для прямоугольного раскройного плана, избегающей пересечения имеющихся резов // Proceedings of the 3-rd International Conference «Information Technologies for Intelligent Decision Making Support». 2015. Vol. 1. P. 39–43.

90. *Макаровских Т.А., Панюков А.В., Савицкий Е.А.* Алгоритмы маршрутизации для систем технологической подготовки процессов раскроя // Системы проектирования технологической подготовки производства и управления этапами жизненного цикла промышленного продукта(CAD/CAM/PDM-2015). Труды 15-й международной конференции. Под ред. А.В. Толока. – М.: ООО «Аналитик». – 2015. С. 182–186.
91. *Макаровских Т.А., Панюков А.В.* Алгоритм построения АОЕ-цепи в плоском связном 4-регулярном графе // Материалы XII Международного научного семинара «Дискретная математика и ее приложения» имени академика О. Б. Лупанова. – М.: Издательство механико-математического факультета МГУ. 2016. С. 293–296.
92. *Макаровских Т.А., Панюков А.В.* Определение траектории режущего инструмента с отсутствием пересечения резов // Системы проектирования технологической подготовки производства и управления этапами жизненного цикла промышленного продукта(CAD/CAM/PDM-2016). Труды 16-й международной конференции. Под ред. А.В. Толока. – М.: ООО «Аналитик». - С. 138–142.
93. *Макаровских Т.А.* Техника программной реализации задачи построения AOE-цепи в плоском 4-регулярном графе // тр. Пятой Междунар. науч. конф., Банное, Россия, 24–28 февр. 2016 г. (ИТиС–2016) : науч. электрон. изд. / отв. ред. Ю. С. Попков, А. В. Мельников. Челябинск : Изд-во Челяб. гос. ун-та, 2016. – С. 17–20.
94. *Makarovskikh T.A., Savitskiy E.A.* The features of cutting plans design for different cutting technologies// Information Technologies for Intelligent Decision Making Support (ITIDS'2016) Proceedings of the 4th International Conference. 2016. C. 98–103.
95. *Макаровских Т.А., Савицкий Е.А.* Построение АОЕ-цепи в плоском графе// Дискретная математика, алгебра и их приложения: Тез. докл. Межд.науч. конф. Минск, 14–18 сентября 2015 г. – С. 44–45.
96. *Макаровских Т.А.* О мощности эйлерова ОЕ-покрытия плоского графа// Информационные технологии интеллектуальной поддержки принятия решений (Proceedings of the 5th International Conference «Information Technologies for Intelligent Decision Making Support» and

the Intended International Workshop «Robots and Robotic Systems»). 2017. С. 170–174.

97. *Макаровских Т.А., Панюков А.В.* Алгоритм построения самонепересекающегося ОЕ-маршрута в плоском графе // тр. Шестой Междунар. науч. конф., Банное, Россия, 1–5 марта 2017 г. (ИТиС — 2017): науч. электрон. изд. / отв. ред. Ю. С. Попков, А. В. Мельников. Челябинск: Изд-во Челяб. гос. ун-та, 2017. – С. 157–162.
98. *Макаровских Т.А.* Способ решения задачи маршрутизации на основе построения покрытий в плоских графах// Перспективные информационные технологии (ПИТ–2018) Труды Международной научно-технической конференции. Под редакцией С.А. Прохорова. 2018. С. 831–835.
99. *Макаровских Т.А., Панюков А.В., Савицкий Е.А.* Программное обеспечение для задачи построения траектории движения режущего инструмента // Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта (CAD/CAM/PDM–2018) Труды XVIII Международной молодёжной конференции. Под общей редакцией А.В. Толока. 2018. С. 172–176.
100. *Makarovskikh T., Panyukov A., Savitsky E.* Software Development for Cutting Tool Routing Problems // Procedia Manufacturing, 2019. Vol. 29. P. 567–574. DOI: 10.1016/j.promfg.2019.02.123
101. *Makarovskikh T., Savitsky E.* Optimization on Pierce Points Number for Cutting Plan with Combined Cuts // XVIII International Conference "Mathematical Optimization Theory and Operations Research". Abstracts. – Ekaterinburg, Russia: Publisher "UMC UrFU 2019. – p. 114.

Работа выполнялась в соответствии с планами госбюджетных НИР ЮУрГУ (номер гос. регистрации 01.2001 05137). Работа поддерживалась грантами РФФИ «Урал» (проекты 01-01-96401, 10-07-96002-р_урал_а), Грантом Президента РФ МК-2603.2008.9, Губернаторским грантом Челябинской области р2001урчел-01-04 и грантами губернатора Челябинской области для студентов, аспирантов и молодых ученых в 2002, 2003, 2013 гг. Часть работы выполнялась в рамках соглашения № 14.B37.21.0395 с Министерством образования и науки Российской Федерации от 06 августа 2012 года.

Формат 60 × 84 1/16. Бумага офсетная. Печать офсетная. Авт. л. 2,0. Уч.-изд. л. 2,4.
Подписано в печать __.__.2020. Тираж 100 экз.
Издательский центр ЮУрГУ. 454080, г. Челябинск, пр. Ленина, 76