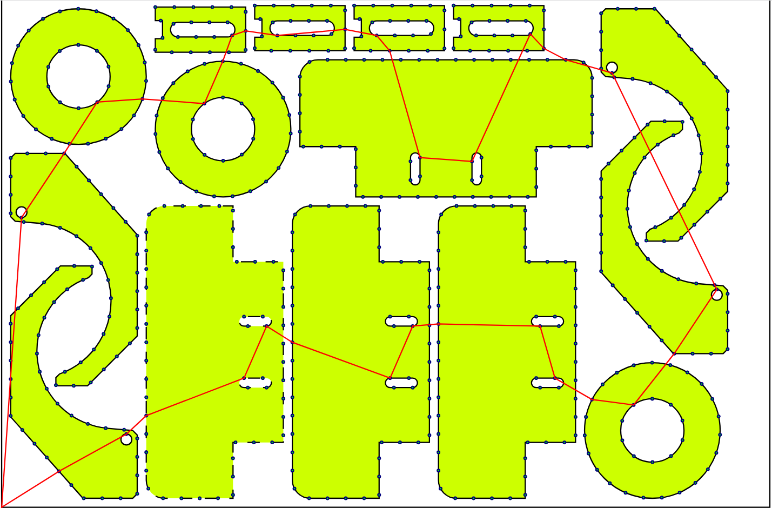
Ответы на вопросы / замечания официального оппонента д.т.н., профессора **Верхотурова Михаила Александровича**

1. *В диссертационной работе использовано несколько языков программирования – С, Julia, Python, JavaScript. Чем вызвано такое разнообразие? Каковы причины выбора этих языков программирования?*

В значительной степени были продолжено использование тех языков, которые уже использовались в исследованиях, проводимых в нашей научной школе, хотя есть и объективные основания для выбора. Язык Python, например, использовался для решения задачи PCGTSP из-за наличия (и опыта использования) специализированных библиотек для работы с графами и матрицами. Язык C подошёл для решения задачи CCP благодаря хорошей производительности и простоте – эта эвристика не требует сложных структур данных. Использование языка JavaScript мотивировано прежде всего его доминированием (хотя уже и не абсолютным) в современных браузерах.

1. *Численные эксперименты второй главы (решение задачи PCGTSP) проводились на открытой библиотеке PCGTSPLIB, содержащей абстрактные экземпляры обобщённой задачи коммивояжера с ограничениями предшествования, зачастую асимметричные. В то же время автор принимал участие в разработке библиотеки экземпляров задач именно маршрутизации инструмента, где матрица расстояний и частичный порядок определяются геометрией плоских деталей. Проведение численных экспериментов над этой библиотекой дало бы более наглядные результаты и лучше бы продемонстрировало возможность применения алгоритма для проектирования управляющих программ машин листовой резки с ЧПУ.*

Согласен с замечанием. На момент разработки алгоритмов в этой библиотеке ещё не было примеров интересной размерности (30–40 контуров). Поэтому алгоритм решения задачи PCGTSP, конечно, тестировался на примерах из библиотеки (небольшой размерности), но основное исследование выполнялось на библиотеке PCGTSPLIB, в том числе для сравнения с другими исследованиями, опирающимися на ту же библиотеку. Сейчас задачи на резку добавлены в библиотеку и планируется проведение численных экспериментов для сравнения всех задействованных в работе алгоритмов.

1. *Вопросы оптимизации раскроя намеренно не включены в круг исследуемых в диссертационной работе. Тем не менее, интересны перспективы применения описанных в алгоритмах для совместной оптимизации процессов раскроя и резки плоских деталей.*

Да, задача совместной оптимизации раскроя и резки INRP (Integrated Nesting & Routing Problem) представляет большой теоретический и практический интерес, являясь ещё более сложной, чем задачи раскроя и резки по отдельности. Полагаю, что разработанные алгоритмы и модели могут применяться решении этой задачи в дальнейших исследованиях. В диссертации приведён пример применения непрерывно-дискретной модели для решения задачи сегментной резки.

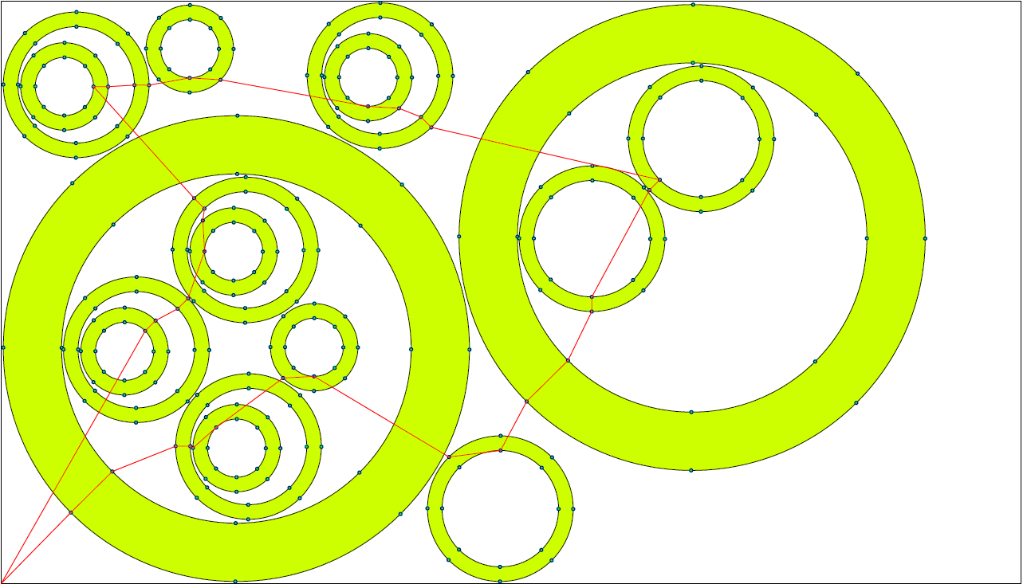
1. *Из текста диссертации непонятна методика получения оценок различных методов построения нижней оценки в табл. 2.2 на стр. 44*

Согласен с замечанием. Для получения этих оценок использовались как раз раскройные планы из библиотеки задач резки сравнительно небольшого размера (например, e5x\_1 с 23 кластерами), для них выполнялся полный алгоритм ветвей и границ со всеми вариантами расчёта и собиралась статистика по полученным оценкам для каждого варианта. Распределение оценок оказывается, как и ожидалось, близким к нормальному, по этим распределениям и получены цифры в указанной таблице.

1. *Чем можно объяснить использование диссертантом метода обхода в ширину на стр. 48, а не, например, обхода в глубину?*

При обходе в глубину (например, в оригинальной работе Салмана), алгоритм ветвей и границ быстро находит решение (не обязательно оптимальное) и может его использовать для отсечения ветвей дерева поиска, при этом нижние оценки не получаются или получаются значительно позже. В нашем случае первоначальное решение получается эвристикой PCGLNS, поэтому отсечение работает с самого начала и применяется обход в ширину, чтобы сразу же начать получать нижние оценки и постепенно уточнять их по мере работы алгоритма.

1. *Можно ли сравнить эффективность получения точного решения задачи GTSP алгоритмами, разработанными А. Г. Ченцовым, и алгоритмами, предложенными автором диссертации?*

Алгоритм А. Г. Ченцова всегда находит точное решение, однако он работает не для всех задач, существует граница сложности, максимальное количество контуров (порядка 33). Алгоритм ветвей и границ способен находить точные решения для некоторых из таких задач, это было одной из мотиваций к его разработке. На рис. 2.1 на стр.  38 как раз приведён эмпирически подобранный пример (34 контура), который (на момент исследования) не решался алгоритмом А. Г. Ченцова, но легко решался алгоритмом ветвей и границ. Кроме того, он даёт приближенное решение и нижнюю оценку возможных решений. Если сравнивать быстродействие (для тех задач, которые решаются обоими алгоритмами), то алгоритм А. Г. Ченцова в общем работает заметно быстрее. Конкретные цифры будут получены после проведения соответствующего исследования, которое уже запланировано.

Ответы на вопросы / замечания официального оппонента д.т.н., профессора **Коновалова Анатолия Владимировича**

1. *В описании численных экспериментов для алгоритма решения задачи PCGTSP (вторая глава) указано, что они проводились на вычислительном кластере «Уран» института математики и механики им. Н. Н. Красовского Уральского отделения Российской академии наук и приведены его характеристики – 16-ядерные Intel Xeon, 128G RAM. На каком оборудовании проводились численные эксперименты третьей главы с задачей непрерывной резки?*

Данный алгоритм в настоящее время реализован только под ОС Windows и не может исполняться на кластере. Расчёты велись на обычных персональных компьютерах и ноутбуках, например 4-ядерный Intel Core i7 CPU, 8 Gb RAM, операционная система Windows 7.

1. *При разработке управляющих программ для оборудования термической резки с ЧПУ важным является учёт тепловых деформаций, возникающих в процессе резки. Каким образом возможно реализовать такой учёт в описанных в диссертационной работе алгоритмах?*

Полагаю, что это возможно. На данный момент существуют два основных подхода к учёту тепловых деформаций – 1) использование эмпирических правил типа «жёсткости детали» и «жёсткости листа», опирающиеся на геометрическим соображения, и 2) прямой расчёт тепловых полей, возникающих в процессе резки. Второй подход значительно более трудоёмкий, он доминирует в исследованиях зарубежных коллег. У нас он также исследовался, но скорее для подтверждения первого подхода, который уже реализован в программном обеспечении Routing Manager, использующим алгоритмы А. Г. Ченцова и П. А. Ченцова. Задача реализации его в новых алгоритмах представляется скорее технической, чем научной.

1. *В доказательстве утверждения 3.1 на стр. 68 функция φ(t) названа выпуклой, во избежание двусмысленности следовало бы уточнить, что она является выпуклой вниз, а не вверх, что подтверждается тем, что её вторая производная неотрицательна.*

Согласен с замечанием, имелась в виду функция, выпуклая вниз.

1. *В тексте диссертации списки оформлены вразнобой, иногда пункты списков начинаются со строчной буквы, иногда с прописной, в конце ставится иногда точка, иногда точка с запятой, а иногда ничего.*

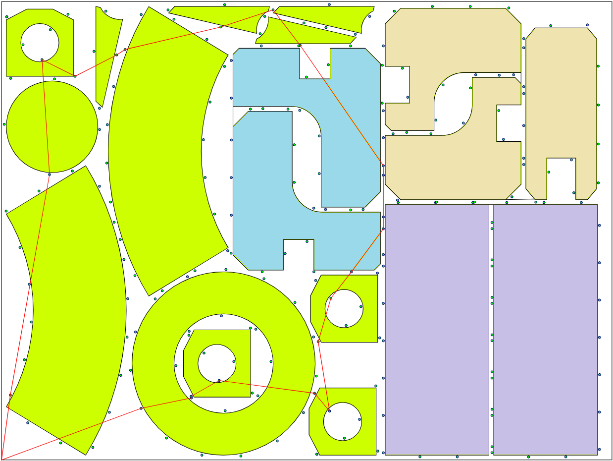
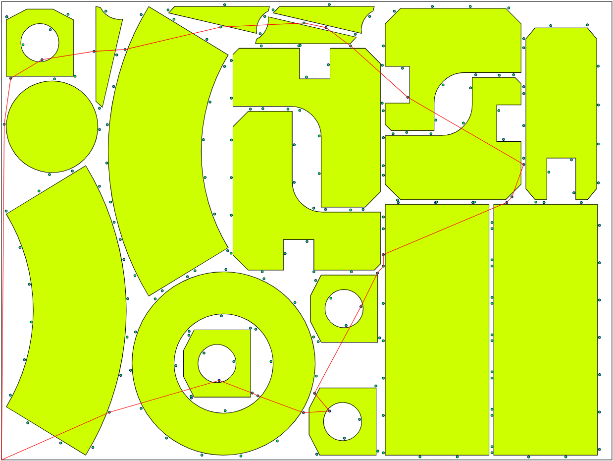
Согласен с замечанием.

1. *В тексте диссертации также имеются опечатки, так в формуле на стр. 18 вместо CJ должно быть Cj по смыслу.*

Согласен с замечанием. Хотя при подготовке текста диссертации использовались средства проверки правописания, отловить все ошибки они не в состоянии, в особенности опечатки в математических формулах.

Ответы на вопросы / замечания официального оппонента д.т.н., доцента **Ложникова Павла Сергеевича**

1. *Во введении (раздел 1.1) описаны различные техники резки, применяемые в современном производстве, а именно: стандартная, мультиконтурная и мультисегментная. В то же время, описанные алгоритмы ориентированы на использование только резки по замкнутому контуру, то есть стандартной. Каким образом возможно использовать нестандартные техники резки в рамках разработанного в диссертации алгоритмического обеспечения для решения задач оптимальной маршрутизации инструмента?*

Мультиконтурная резка может применяться в обоих разработанных алгоритмах без какой-либо их модификации. На рисунке изображен пример мультиконтурной резки, полученной алгоритмом решения задачи PCGTSP с использованием перемычек (мостиков). Мультисегментная резка напрямую в этих алгоритмах вестись не может, требуется некоторая их доработка.

1. *Алгоритм решения задач PCGTSP, описанный в Главе 2, решает их в самой общей постановке, для произвольных входных данных, тогда как задачи, возникающие при оптимизации маршрута резки, существенно эвклидовы. Каким образом учитывается это их свойство в алгоритме? Возможно ли использовать геометрические соображения для улучшения работы алгоритма?*

Непосредственное использование геометрических соображений в текущем дизайне алгоритма вряд ли возможно, потому что исходный файл задания очищен от геометрической информации. Тем не менее, по результатам исследований в алгоритм был добавлен механизм релаксации H2 на основе цепей длины 2, как описано на стр. 42. Он позволяет точнее учитывать взаимное расположение контуров на плоскости, то есть косвенно использует их геометрию. Примерно в половине случаев этот механизм приводит к улучшению нижней оценки. Вообще же, на мой взгляд использование геометрических соображений может оказаться чрезвычайно полезным для получения более точных оценок.

1. *Если результаты работы алгоритма сопоставимы с результатами, полученными универсальным решателем Gurobi, то в чём смысл разработки специализированного алгоритма?*

Решатель Gurobi является коммерческим программным продуктом, требующим лицензирования для своего использования. По этой причине в диссертационной работе он использовался как научный инструмент, в качестве базы сравнения, использование его в составе САПР управляющих программ непосредственно на производстве не планировалось.

1. *Английские аббревиатуры систематически вводятся без расшифровки, например САМ на стр. 5 или названия классов задач резки (ССР, ЕСР, ICP и т. д.) на стр. 7. Часть из них приведена в списке сокращений на стр. 97, но не все.*

Частично согласен с замечанием. Аббревиатуры классов задач расшифрованы на стр. 29, но это следовало сделать раньше по тексту.

1. *В тексте работы имеются грамматические ошибки. Так, на стр. 5, 7 и 31 пропущен дефис в термине* *«САD/САМ-система», хотя в других местах он употребляется.*

Частично согласен с замечанием. Конкретно термин «САD/САМ-система» может писаться в обоих вариантах, но в тексте диссертации следовало бы конечно соблюсти единообразие.

Ответы вопросы / замечания по автореферату к.т.н., доцента **Захаровой Галины Борисовны**

1. *На рис. 2 на стр. 10 приведена классическая задача коммивояжера TSP как один из частных случаев задачи резки, хотя как правило последняя сводится к обобщённой задаче коммивояжера GTSP. Не ясно, каким образом простая задача коммивояжера может применяться для поиска оптимального решения задачи маршрутизации режущего инструмента.*

Обобщённая задача коммивояжера вырождается в классическую, если на каждом контуре определена только одна возможная точка врезки. Такой случай приведён в тексте диссертации на стр. 56. Этот класс задач, в частности, рассматривается в работах первого оппонента профессора М. А. Верхотурова.

Кроме того, алгоритм решения задачи непрерывной резки в главе 3 фактически сводит последнюю как раз к классической задаче коммивояжера, только расстояния между городами, роль которых выполняют контуры деталей, задаётся не в виде матрицы, а вычисляется «на лету» в ходе работы эвристики поиска оптимальных позиций точек врезки.

1. *Понятно, что ограничение предшествования сокращает время счёта для алгоритма главы три, не использующего дискретизацию контуров. А как аналогичный эффект достигается для алгоритма ветвей и границ второй главы?*

В двух блоках алгоритма ветвей и границ учёт ограничения предшествования позволяет снизить вычислительную сложность: 1) в момент релаксации исходной задачи PCGTSP в задачу ATSP и 2) в процедуре ветвления. В обоих случаях в ходе исследования удалось увеличить эффективность отсечения (соответственно рёбер графа и ветвей дерева), по сравнению с подходами, описанными в литературе.

1. *Не для всех реализованных в работе алгоритмов приведено время счета. Хотелось бы получить представление о сравнительном быстродействии этих алгоритмов.*

Согласен с замечанием. Алгоритмы тестировались в разное время и на разных задачах. Первым был разработан алгоритм решения задачи непрерывной резки и нас интересовала его работоспособность. Сейчас, с появлением библиотеки задач резки, планируется провести численный эксперимент по сравнению всех задействованных алгоритмов.

Ответы на вопросы / замечания по автореферату д.т.н., профессора **Мельникова Андрея Витальевича**

1. *В автореферате при описании разработанных в работе алгоритмов решения задач маршрутизации говорится только об ограничениях типа условий предшествования, при этом не обговаривается учёт других технологических ограничений термической резки.*

Согласен с замечанием. Учёт других ограничений термической резки не входил в задачи исследования. Для практического применения разработанных алгоритмов именно для термической резки, требуется их доработка. Поскольку этот вопрос уже решён на практике для алгоритма А. Г. Ченцова и П. А. Ченцова, эта доработка будет носить скорее технический характер.

1. *На стр. 18 упомянуто, что алгоритм, использующий модель непрерывно-дискретной оптимизации, сравнивается с точным алгоритмом* *А. Г. Ченцова, разработанным на основе метода динамического программирования. В дополнение к этому следовало бы привести сравнение и с алгоритмом ветвей границ, разработанным в диссертационной работе. Имеется в виду и факт получения глобального экстремума и вычислительные затраты на его получение.*

Согласен с замечанием. В момент исследований алгоритма непрерывной резки алгоритм решения задачи PCGTSP ещё не существовал, именно поэтому для сравнения был использован алгоритм А. Г. Ченцова. Впрочем, для конкретных примеров, приведённых в автореферате и диссертации, на которых тестировалась задача CCP, оба алгоритма дают одинаковые маршруты. С появлением библиотеки задач резки стало возможным провести полноценное сравнение всех задействованных алгоритмов, и такой эксперимент сейчас готовится.

Ответы на вопросы / замечания по автореферату д.т.н., профессора **Мартынова Виталия Владимировича**

1. *К недостаткам автореферата можно отнести почти полное отсутствие описания деталей программной реализации алгоритмов, приведён псевдокод только одного программного блока*

Не согласен с оценкой. Объём автореферата не позволяет приводить исходные коды программ в значительном количестве. В тексте диссертации приводится псевдокод ещё нескольких других программных блоков, хотя и не покрывающих разработанные алгоритмы целиком. Полный исходный код алгоритма ветвей и границ вместе с исходными данными и результатами численных экспериментов выложен в репозиторий git на сайте GitHub, и находится в свободном доступе. Ссылка на репозиторий приведена в тексте диссертации.

1. *С точки зрения оформления неудачно выбрано расположение в тексте рисунка 1 на стр. 8.*

Согласен с замечанием. Для подготовки текста диссертации и автореферата использовалась система компьютерной вёрстки LaTeX, в которая расположение рисунков и таблиц в тексте определяется внутренними алгоритмами системы.

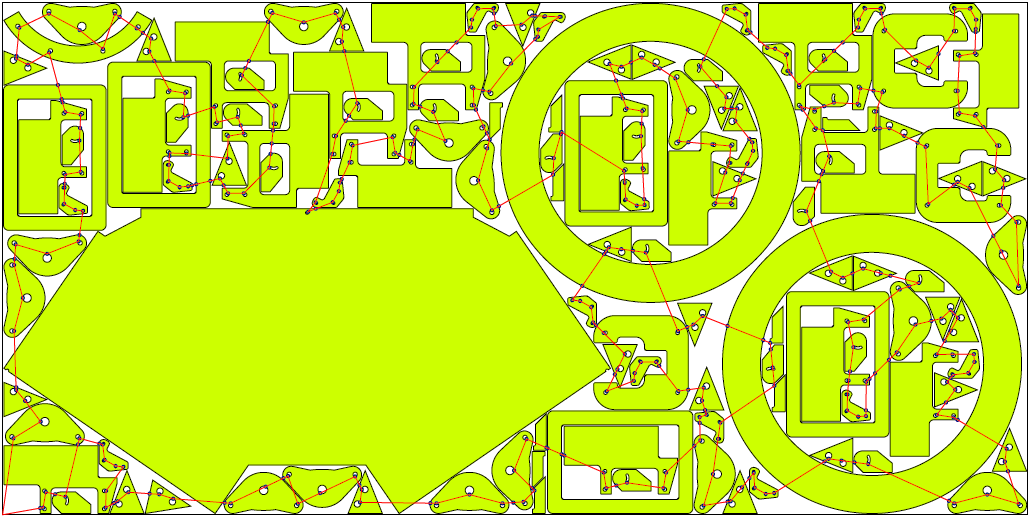
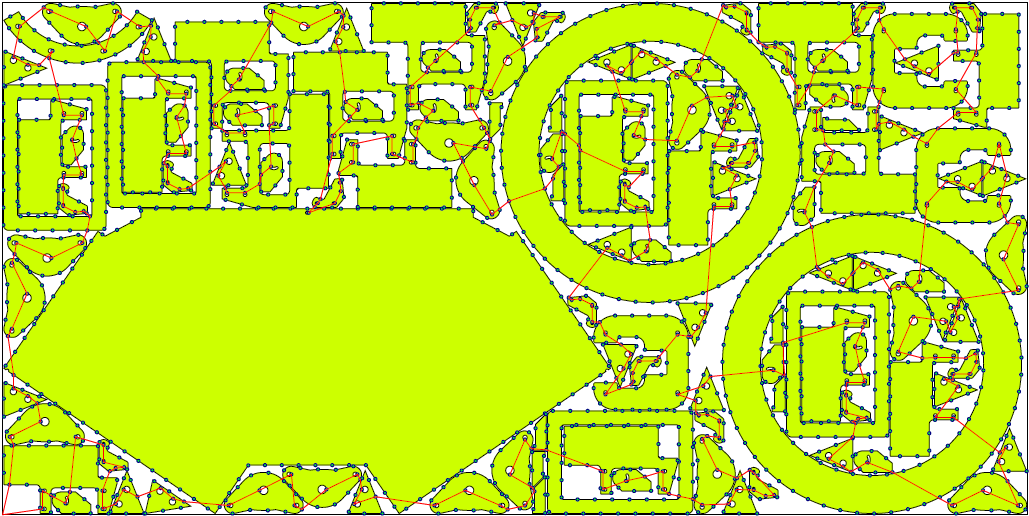
1. *Сформулированные основные результаты и выводы по работе не коррелируют однозначно с поставленными задачами.*

Частично согласен с замечанием, хотя различия носят, на мой взгляд, стилистический характер. Задачи исследования ставились в общем виде, а результаты описаны более подробно. Так, результаты 1–4 относятся к первой задаче исследования, результаты 4–6 – ко второй, результаты 8–9 к четвёртой задаче, и результат 10 однозначно соответствует третьей задаче исследования.

Ответы на вопросы / замечания по автореферату к.т.н. **Таваевой Анастасии Фидагилевны**

1. *Хотелось бы сравнить результаты работы алгоритмов второй и третьей глав на одних и тех же раскройных планах, что позволило бы лучше судить о применимости этих алгоритмов для разработки управляющих программ для машин листовой резки с ЧПУ.*

Отчасти этот вопрос освещён в тексте диссертации на стр. 54 и 76, где приведены результаты работы обоих алгоритмов для задач большой размерности (423 и 620 контуров).



Видно, что в значительной части маршруты резки, полученные разными алгоритмами, совпадают. В задачах меньшей размерности совпадение как правило полное. При этом длина полученного маршрута как правило немного короче для случая непрерывно-дискретной оптимизации, что легко объяснимо. Разница может составлять от 1% до 10% для сложных раскройных планов.

1. *В автореферате отсутствуют экспериментальные данные о быстродействии реализации алгоритма, использующего непрерывно-дискретную оптимизацию.*

Согласен. На это замечание уже дан комментарий выше.

1. *Интересно было бы исследовать влияние уровня вложенности деталей в раскройном плане на быстродействие описанных алгоритмов.*

Согласен с замечанием. Уровень вложенности оказывает существенное влияние на производительность алгоритмов, и это влияние отчётливо проявляется в численных экспериментах, проведённых в ходе диссертационной работы. Насколько я знаю, А. Г. Ченцов и Я. В. Салий проводили такое исследование для некоторых вариантов алгоритмов маршрутизации, но для разработанных алгоритмов это следует исследовать отдельно.