

Московский авиационный институт
(Национальный исследовательский университет)

На правах рукописи

УДК 515.2

Ньи Ньи Хтун



**РАЗРАБОТКА И ИССЛЕДОВАНИЕ РЕЦЕПТОРНЫХ
ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ ТЕЛЕСНОЙ ТРАССИРОВКИ**

Специальность: 05.01.01 – Инженерная геометрия и компьютерная графика

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель:
кандидат технических наук,
проф. Л.В.МАРКИН

Москва 2014

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1 АНАЛИЗ МЕТОДОВ ПРОЕКТИРОВАНИЯ СОЕДИНИТЕЛЬНЫХ ТРАСС В ЗАДАЧАХ АВТОМАТИЗИРОВАННОЙ КОМПОНОВКИ	16
1.1 Методы проектирования и геометрические модели каналových поверхностей.....	16
1.2 Основные виды геометрических моделей компокуемых объектов	26
1.3 Основные геометрические модели размещения компокуемых объектов в пространстве	31
ЗАКЛЮЧЕНИЕ И ВЫВОДЫ К ГЛАВЕ 1	38
2 РАЗРАБОТКА И ИССЛЕДОВАНИЕ РЕЦЕПТОРНЫХ ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ В ЗАДАЧАХ ТЕЛЕСНОЙ ТРАССИРОВКИ.....	40
2.1 Физическая постановка задачи исследования.....	40
2.2 Математическая постановка задачи исследования.....	47
2.3 Анализ исследований в области автоматизации проектирования трасс.....	49
2.4 Обоснование выбора метода геометрического моделирования телесной трассировки.....	52
2.5 Выбор направления разработки алгоритмов телесной трассировки.....	57
ЗАКЛЮЧЕНИЕ И ВЫВОДЫ К ГЛАВЕ 2	61
3 РАЗРАБОТКА И ИССЛЕДОВАНИЯ РЕЦЕПТОРНЫХ ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ ФОРМИРОВАНИЯ КАНАЛОВЫХ ПОВЕРХНОСТЕЙ	63

3.1 Исследование и анализ известных алгоритмов трассировки, основанных на рецепторных геометрических моделях.....	63
3.2 Выбор направлений модификации известных алгоритмов трассировки, основанных на рецепторных геометрических моделях	76
3.3 Разработка геометрической модели построения главной направляющей линии (ГНЛ) среди уже размещенных объектов	80
3.4 Разработка эвристик для построения главной направляющей линии канала.....	90
ЗАКЛЮЧЕНИЕ И ВЫВОДЫ ПО РАЗДЕЛУ 3.....	98
4 АЛГОРИТМИЧЕСКАЯ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННОЙ ГЕОМЕТРИЧЕСКОЙ МОДЕЛИ ТЕЛЕСНОЙ ТРАССИРОВКИ.....	100
4.1 Алгоритмическая реализация предложенной геометрической модели телесной трассировки.....	100
4.2 Особенности программной реализации предложенной геометрической модели телесной трассировки	125
4.3 Интерфейс программы, реализующий алгоритм трассировки	131
4.4 Исследование и верификация алгоритма и программы телесной трассировки.....	136
ЗАКЛЮЧЕНИЕ И ВЫВОДЫ ПО РАЗДЕЛУ 4.....	148
ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ ПО РАБОТЕ.....	150
СПИСОК ЛИТЕРАТУРЫ	153
Приложение 1	179
Приложение 2.....	180

ВВЕДЕНИЕ

Актуальность исследования.

При автоматизации проектирования любой техники на результат проектирования оказывает существенное влияние качество компоновки. Исследование методов и моделей автоматизированной компоновки проводится достаточно давно. При этом одной из задач компоновки является решение задач трассировки, т.е. проектирования коммуникаций между уже размещенными объектами (рисунок В.1). Такие задачи, связанные, например, с задачами прокладки электрожгутов между компоновемыми объектами, являются достаточно трудноформализуемыми и сложными для решения из-за присущей им многоэкстремальной природы.

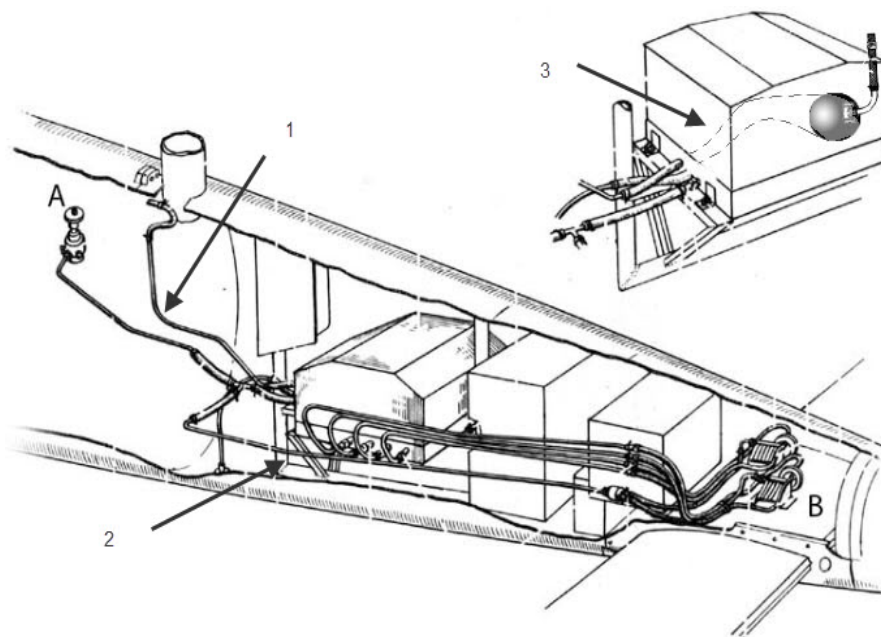


Рисунок В.1 – Проведение коммуникаций между заданными точками пространства с учетом областей запрета в транспортном средстве

Особым и значительно более сложным видом трассировки является так называемая «телесная» трассировка, т.е. такой случай, когда размеры (соединительных элементов) трассы сопоставимы с размерами компоновемых элементов. На

практике, это проектирование трубопроводов, воздухопроводов и других элементов транспортных систем и, прежде всего, авиационной техники (рисунок В.2).



Рисунок В.2 – Случай “телесной” трассировки

Проблема построения заданной топологии электрической сети достаточно успешно решена для разводки электрических проводов (например, в печатных платах). Это стало возможным потому, что к топологии сети не предъявляется никаких дополнительных требований, кроме обеспечения заданной конфигурации соединения проводников с навесными элементами. Однако выдвижение дополнительных требований к трассировке резко усложняет геометрическое и математическое описание поставленной задачи.

Долгое время нахождение пути трассы было традиционной задачей искусственного интеллекта, с использованием которого было предложено несколько вариантов ее решений. Проблема построения рациональной (даже не оптимальной) трассы актуальна в огромном спектре задач: от компьютерных игр до расчета тра-

ектории роботов и вездеходов для перемещения в пространстве, от построения дорог и управления перевозками до проектирования автоматизированных систем.

Усложнение проектируемой техники в аэрокосмической промышленности, транспорте, строительстве и других сферах жизни приводит к усложнению построения и оптимизации подключения сетей в областях сложной геометрической формы, которые геометрически не могут быть рассмотрены просто как связанные многообразия. Специфика этих задач состоит в том, что на проектирование трасс накладываются дополнительные различные конструктивные и технологические требования. Например, при проектировании гидро- и пневотрасс часто накладываются определенные дополнительные ограничения по гладкости трассы или же обеспечения минимально допустимого расстояния между уже размещенными объектами (для обеспечения требований электрической или пожарной безопасности).

Степень разработанности объекта исследования.

Некоторые из описанных задач могут быть решены известными геометрическими методами, полученными в ранее проведенных исследованиях в этой области. Однако подавляющее большинство исследований в этой области посвящены проектированию каналовой поверхности (пусть даже высокого порядка гладкости) без учета каких-либо других уже размещенных объектов, которые могут “не позволить” осуществиться столь замечательной задумке конструктора.

Именно специфика проведения трассировки среди уже размещенных объектов намного усложняет задачу трассировки как геометрически, так и математически. Это связано с тем, что задачи автоматизации компоновки даже объектов простейшей геометрической формы отличаются наиболее низкой степенью формализации. Что же говорить, когда предметом компоновки являются объекты сложных геометрических форм, на размещение которых накладываются дополнительные конструктивные и технологические требования.

Все это позволяет нам утверждать, что многие из вышеперечисленных задач автоматизированной компоновки (в частности наиболее общий случай – телесная трассировка среди уже скомпонованных объектов с учетом заданной гладкости линии тока) пока еще ждут своего решения и являются предметом исследования данной диссертации.

Цели и задачи диссертации:

Целью данной работы является поиск наиболее эффективной (с учетом заданных ограничений) трассы от заданной начальной точки к конечной точке с учетом областей запретов. Предполагается, что размеры трассы соизмеримы с размерами уже размещенных объектов (случай телесной трассировки). Решение поставленной задачи предусматривает:

1. Разработку геометрических моделей проектирования соединительных трасс как размещаемых объектов среди уже размещенных с учетом дополнительных конструктивных и технологических требований (заданных точек входа и выхода трассы, заданных поперечных сечений или закона их изменения, заданного минимального радиуса кривизны трассы и заданного минимального расстояния прохождения трассы от уже размещенных объектов).

2. Разработку математического и программного обеспечения для реализации разработанной геометрической модели проектирования трассы с дополнительными заданными конструктивными и технологическими ограничениями.

3. Исследование и верификация разработанных геометрических моделей.

4. Внедрение полученных результатов в процесс реального проектирования и учебный процесс.

Научная новизна диссертации заключается в решении следующих задач и формулировании новых научных положений:

1. Сформулирована физическая и математическая постановка задачи автоматизированной телесной трассировки как многокритериальная задача математического программирования.

2. Показана перспективность использования рецепторного метода геометрического моделирования для решения поставленной задачи – компоновки таких сложных по своей геометрической форме объектов как каналовые поверхности.

3. Показана невозможность использования даже лучших известных алгоритмов дискретной трассировки (алгоритма Дейкстры и алгоритма A^*) для автоматизации трассировки каналовых поверхностей.

4. Разработана геометрическая модель и алгоритм построения главной направляющей линии каналовой поверхности для плоской и пространственной трассы, являющийся глубокой модернизацией алгоритма A^* и устраняющий основные ограничения прототипа A^* - возможность прокладки плавных трасс на заданных расстояниях от областей запрета.

5. Разработаны эвристики, повышающие эффективность работы алгоритма трассировки, направленные на выбор рациональных направлений движений к следующей точке будущей траектории.

6. Для предложенного алгоритма разработано реализующее его программное обеспечение на языке C#, обеспечивающего средствами интерфейса программы настройку режимов и параметров трассировки, а также визуализацию полученных компоновочных решений.

7. Проведена оптимизация информационной структуры алгоритма для повышения скорости работы программы, позволившая увеличить ее быстродействие по сравнению с ближайшими аналогами в 300 -1200 раз.

8. Проведена оценка точности представления телесной трассы рецепторной матрицей, показавшей на тестовых примерах, что погрешность представления зависит от размера рецептора d и составляет примерно $0,9 d \pm 0,28d$ при доверительном интервале $\pm 3\sigma$.

9. Проведена оценка производительности реализации предложенного рецепторного алгоритма, показывающего, что процессорное время расчета компоновки

возрастает примерно по параболической зависимости от количества рецепторов на единице длины рецепторной матрицы, но все равно составляет доли секунды.

10. Проведено с помощью предложенного метода трассировки исследование возможности прокладки воздуховода в моторном отсеке легкого самолета “АСА-2”. Результаты исследования также внедрены в учебный процесс кафедры инженерной графики МАИ.

Теоретическая и практическая значимость работы заключается в разработке на основании предложенной геометрической модели телесной трассировки с учетом дополнительных конструктивных и технологических факторов алгоритмического и программного обеспечения, позволяющего:

- Проектировать на плоскости и в пространстве в автоматическом режиме соединительные трассы с учетом заданных точек входа и выхода трассы, заданного минимального радиуса кривизны и площади сечения, а также заданного минимального расстояния от областей запрета и других скомпонованных объектов;

- Провести исследование, верификацию и тестирование разработанного алгоритмического и программного обеспечения;

- Внедрить разработанное алгоритмическое и программное обеспечение в практику проектных исследований на легком самолета “АСА-2” и в учебный процесс МАИ;

- Наметить пути совершенствования разработанного алгоритмического и программного обеспечения в существующие САД системы как автономного расчетного модуля.

Методология и методы исследования.

Объектом исследования являются геометрические модели и алгоритмы автоматизированного размещения соединительных трасс с учетом уже размещенных объектов и заданных конструктивных и технологических ограничений на прохождение трассы.

Предметом исследования являются геометрические модели и алгоритмы анализа взаимного положения уже размещенных (ранее скомпонованных объектов) и возможных путей прокладки трассы с учетом дополнительных конструктивных и технологических ограничений.

Методологическую основу работы составляют методы геометрического и математического моделирования, классические методы математического программирования, дискретного анализа и теории множеств, теории графов, теории алгоритмов. В математической постановке задача телесной трассировки представляется как задача многокритериальной дискретной оптимизации

Методологические и теоретические основы исследования основаны на фундаментальных трудах в области:

- метода геометрического моделирования, изложенные в работах отечественных ученых Валькова К.И. [19], Денискина Ю.И. [39], Иванова Г.С. [63], Котова И.И. [78, 79], Михайленко В.Е. [103, 104, 105], Осипова В.А. [129, 130], Полозова В.С. [141], Похвалинского А.Б. [144], Рвачева В.Л. [150, 151], Рыжова Н.Н. [158, 159, 160], Стародетко Е.А. [180], Фролова С.А. [196], Якунина В.И. [209] и др, а также зарубежных Поля де Кательджо, Роберта Фергюсона (Robert Fergusson), Стивена Кунса (Steven Coons), Пьера Безье (Pierre Bézier), Шарля Эрмитта (Charles Hermite), Исаака Шенберга (Isaac Jacob Schoenberg), Карла де Бура (Carl de Boor), Кена Версприлла (Ken Versprille), Юджина Ли (Eugene Lee), Стива Гейзберга (Steve Ginsberg) и других.

- методов геометрического моделирования каналových поверхностей и дифференциальной геометрии, изложенных в работах Осиповым В.А. [131, 132] и его учеников - Андреева В.А. [5], Зелева В.П. [54], Лелюшенко С.И. [86], Мезенцева Л.Г. [99], Поликарпова Ю.В. [140], Миролюбовой Т.И. [101, 102], а так же другими исследователями московской и общероссийской геометрической школы - Александровича В.П. [3], Дёминой В.А. [41], Иванова Г.С. [61, 62, 63], Кулиша А.С. [82], Хасанова В.Х. [197] и др. Следует отметить большой вклад в решение

этого вопроса Киевской научной школы прикладной геометрии - труды Блюка А.В. [16], Василевского О.В. [20], Грибова С.М. [35], Дорошенко Ю.А. [42, 43], Кожушко Н.А. [71], Пилипаки С.Ф. [139], Радзивилловича В.В. [149], Обуховой В.С. [126, 127], Скидана И.А. [172] и др.

- общей методики автоматизации проектирования, изложенной в трудах Горелика А.Г. [31], Норенкова И.П. [115, 116, 117], Сидоренко С.М. [168], Вермишева Ю.Х. [21, 22], Падалко С.Н. [133], Петренко А.И. [137], Советовой Б. Я. [176], Прохорова А.Ф. [146], Курейчика В.М. [84], Ли К. [85], а также ряда заграничных авторов - Гардана И. (Yvon Gardan) и Люка М. (Michel Lucas) [26], Грувера М. (Mikell P. Groover) и Зиммерса Э. (Emory W. Zimmers) [37], Гиллоя В. (Wolfgang K. Giloi) [30], Принса М. [147], Шенена П. (Peter Shenen) [97], Шпура Г. (Gunter Spur) и Краузе Ф. (Frank-Lothar Krause) [203], Энгельке У. (William D. Engelke) [204], Хокса Б. (Barry Hawkes) [198], Хорафаса Д., (Dimitris N. Chorafas) и Легга С. (Stephen J. Legg) [199] и других.

- методики автоматизации проектирования авиационной техники, изложенная в трудах Волошина В.В. [24], Осина М.И. [106, 128], Формалева В.Ф. [195], Мальчевского В.В. [94], Лисейцева Н.К и Самойловича О.С. [46], Куприкова М.Ю. [83], Пухова А.А. [148] и др.

- методики автоматизированного проектирования трасс простейшей формы между уже скомпонованными объектами в трудах Образцова А.А. [125], Брысина В.А. [18], Богацкого И.З. [17], Драганова Б.Х. [44, 45], Искакова С.Д. [67], Кожушко Н.А. [71], Мартыновой О.Г. [98], Некрасовой О.И. [114], Смеляковой С.В., Аристовой И.В. [173, 174, 187] и других.

- методов автоматизации компоновочных работ, изложенных в работах проф. Гаврилова В.Н. [25], проф. Мухачевой Э.А. [107], акад. Рвачева В.Л. [150, 142], проф. Стояна Ю.Г. и Гиля Н.Н. [185, 187, 189], проф. Маркина Л.В. [95] и других, а также зарубежных ученых Bortfeldt A.[213], Cagan J. [215], George J.A.[222], Gilmore P.C.[223], Lim A.[230], Lodi A.[231], Martello S.[233], Pisinger

D.[241], Robinson D. F.[247], Saaty T. L. [249], Szykman S. [252], Vigo D. [254] и др.

- методов автоматизации телесной трассировки трудах Стояна Ю.Г., Аристовой И.В., Смелякова С.В. [8, 186, 9] и Калинина Б.В. [68, 69], а также зарубежными учеными Albano A.[211], Kubiak P[244]. и др.

- методов трассировки больших интегральных схем и печатных плат Абрайтиса Л.Б. [1], Базилевича Р.П. [12, 13], Лузина С.Ю., Петросяна Г.С. и Полубасова О.Б. [87, 88, 89, 90, 92, 142], Петренко А.П. и Тетельбаума А.Я. [138], Селютина В.А. [163], Стешенко В.Б. [181] и др. а также зарубежных ученых Эдсгер Вйбе Дэйкстры (E. W. Dijkstra) [218], Джуда Перл (Judea Pearl) [238,239,240], Ира Поль (Ira Pohl) [242,243], Даниэль Деллинга (Daniel Delling)[217], Питер Харта (Peter E. Hart)[224] и др.

- методов дискретного моделирования геометрических объектов в работах Зозулевича Д.М. [57, 58, 59, 60], Горелика А.Г. [31, 32, 33], Герасименко Е.П. [27, 28, 29], Клишина В.В. [70], Корн Г.В. [74, 75, 76], Рогозы Ю.А. [153, 154], Пашенко О.Б. [136], Ситу Лина [170, 171, 169], а также ряда иностранных авторов – Гаргантини И. (Gargantini I.) [212, 220, 221], Реквишы А.А.Г., (Requcha A.A.G.) [245, 246, 152] и ряда других [220, 221, 237, 249].

Положения, выносимые на защиту:

1. Геометрическая модель телесной трассировки с возможностью построения плавных трасс заданного сечения и заданной минимальной кривизны с обеспечением условия обеспечения заданного расстояния от областей запрета и других скомпонованных объектов.

2. Алгоритм, реализующий геометрическую модель телесной трассировки с использованием дискретной модели пространства (рецепторной модели).

3. Архитектура и программная реализацию алгоритма телесной трассировки, запрограммированную на языке Microsoft C#, обеспечивающую получение компоновочных решений и их визуализацию.

4. Результаты анализа и верификации предложенного алгоритма и его программного обеспечения (оценку точности, быстродействия и др.).

Степень достоверности и обоснованности полученных в работе результатов обеспечивается корректным применением аппарата вычислительной геометрии и компьютерной графики и подтверждается тестированием разработанных геометрических моделей и созданного на их основе программного обеспечения на языке Microsoft C# как при решении тестовых задач с заведомо известным результатом, так и внедрение ее результатов при проектировании воздухопроводов легкого самолета “АСА-2”. Отклонение характеристик расчетных трасс, спроектированных на основе предложенной в диссертации геометрической модели зависят от точности дискретизации компоновочного пространства и при размере рецептора d составляет примерно $0,9 d \pm 0,28d$ при доверительном интервале $\pm 3\sigma$. Результаты теоретических исследований подтверждены публикациями в рецензируемых изданиях и обсуждены на научных конференциях.

Апробация результатов диссертации. Основные положения диссертационной работы докладывались и обсуждались на следующих научно-технических и научно практических конференциях:

1. Инновации в авиации и космонавтике, – М., МАИ, апрель 2010 года.
2. Технологии Microsoft в теории и практике программирования, – М., МАИ. апрель 2010 года.
3. Авиация и космонавтика, – М., МАИ, ноябрь 2010 года.
4. Инновации в авиации и космонавтике, – М., МАИ, апрель 2011 года.
5. Инновации в авиации и космонавтике, – М., МАИ, ноябрь 2012 года.
6. Авиация и космонавтика, – М., МАИ, ноябрь 2012 года.
7. Международный конкурс научных работ по приоритетным направлениям развития науки, технологий и техники в Российской Федерации. – М., МГТУ им. Н.Э.Баумана , 2012 г.

8. 11th International conference on computer applications, – Y., UCSY Yangon, Myanmar, февраль 2013 года.

Содержание диссертационной работы отражено в 11 печатных работах, в том числе в 3-х периодических изданиях, рекомендованных ВАК:

1. Хтун Н. Н., Ситу Л., Маркин Л. В. Рецепторные геометрические модели в задачах автоматизированной компоновки технического отсека легкого самолет // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №47, 2011 г.

URL: <http://www.mai.ru/science/trudy/published.php>

2. Хтун Н. Н., Тайк Ч., Маркин Л. В. Исследование алгоритмов использования рецепторных геометрических моделей в задачах телесной трассировки авиационной техники // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №69, 2013 г.

URL: <http://www.mai.ru/science/trudy/published.php>

3. Хтун Н. Н., Маркин Л. В., Соседко А.А. Применение рецепторных геометрических моделей в задачах автоматизированной компоновки авиационной техники // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №72, 2014 г.

URL: <http://www.mai.ru/science/trudy/published.php>

В других изданиях:

4. Ньи Ньи Хтун. Улучшенный алгоритм трассировки пути, основанный на рецепторном методе // В сб.: Тез. докл. научно – практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2010 ». Тезисы докладов. – М.: Изд-во МАИ. – 2010. – С.108.

5. Ньи Ньи Хтун. Использование рецепторных моделей в задачах трассировки // В сб.: Тез. докл «Технологии Microsoft в теории и практике программирования: труды VII Всероссийской конференции студентов, аспирантов и молодых учёных». – М.: Изд-во Вузовская книга – МАИ. – 2010. – С.85 -86.

6. Ньи Ньи Хтун. Алгоритмы задач трассировки на основе рецепторных геометрических моделей // В сб.: Тез. докл. 9-я Международная конференция «Авиация и космонавтика – 2010 ». – М.: Изд-во МАИ. – 2010 . – С.314-315.

7. Ньи Ньи Хтун. Дискретные модели телесной трассировки // В сб.: Тез. докл (Научно-практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2011») . – М.: Изд-во МЭЙЛЕР. С.190-191.

8. Ньи Ньи Хтун. Использование рецепторного метода для проектирования каналовых поверхностей // в сб.: Тез. докл. научно - практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2012». Тезисы докладов. – М.: Изд-во МАИ. – 2012. – С.256.

9. Ньи Ньи Хтун. Исследование рецепторного метода проектирования каналовых поверхностей в задачах компоновки авиатехники // В сб.: Тез. докл. 11-я Международная конференция «Авиация и космонавтика – 2012». – М.: Изд-во МАИ. – 2012 . – С.298-299.

10. Ньи Ньи Хтун. Использование рецепторных геометрических моделей в задачах телесной трассировки // В сб.: Тру. докл «Международного конкурса научных работ по приоритетным направлениям развития науки, технологий и техники в Российской Федерации., МГТУ им. Н.Э.Баумана». – М.: Изд-во НИИ , 2012. С.146 – 147.

11. Nyi Nyi Htun. Finding the shortest smooth path in variable size using improved A* algorithm on grid-based receptor model // В сб.: Тез. докл. 11-я Международная конференция ICCA 2013 (<http://www.ucsy.edu.mm/ucsy/635558k.do>) « 11th International Conference On Computer Applications – 2013, Yangon, Myanmar ». – М.: Изд-во UCSY. – 2013 . – С.255-260.

1 АНАЛИЗ МЕТОДОВ ПРОЕКТИРОВАНИЯ СОЕДИНИТЕЛЬНЫХ ТРАСС В ЗАДАЧАХ АВТОМАТИЗИРОВАННОЙ КОМПОНОВКИ

1.1 Методы проектирования и геометрические модели каналовых поверхностей

Целью проектирования каналовой поверхности является доставка по определенной траектории некоторого материального носителя (жидкости, газа, электрической энергии от одной точки (точки входа) технического изделия к другой (точке выхода). Для решения поставленной задачи используется два основных подхода к трассировке определяемые используемой метрикой. Метрика - это правило, по которому в данном пространстве определяется расстояние между двумя точками Первый подход – использование Евклидовой метрики. В этом случае трасса проводится в направлении кратчайшего расстояния между точками входа и точками выхода (рисунок 1.1) и длина трассы определяется по теореме Пифагора.

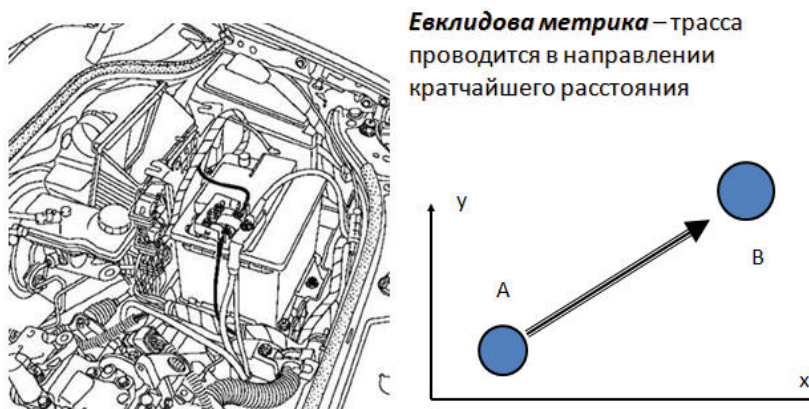


Рисунок 1.1- Проектирование трассы по евклидовой метрике

Вторым подходом является использование Манхэттенской метрики, при которой – трасса проводится в направлении координатных осей (рисунок 1.2). Этот термин введен математиком Германом Минковским и связан с уличной планиров-

кой города Манхэттена). В этом случае расстояние между точками перегиба трассы равно сумме модулей разностей их координат. Трасса в этом случае получается значительно длиннее, чем при использовании Евклидовой метрики, но сам подход дает дополнительные математические возможности для проектирования трассы. Этот подход используется при трассировке больших интегральных схем и печатных план, так как движущиеся по трассе электроны “очень быстрые” и им все равно в какую сторону двигаться. Поэтому резкое изменение траектории и некоторое ее удлинение не оказывает никакого влияния на качество трассировки. Для проектирования же трасс, по которым перемещаются газы или жидкости этот тип метрики практически не применяется.

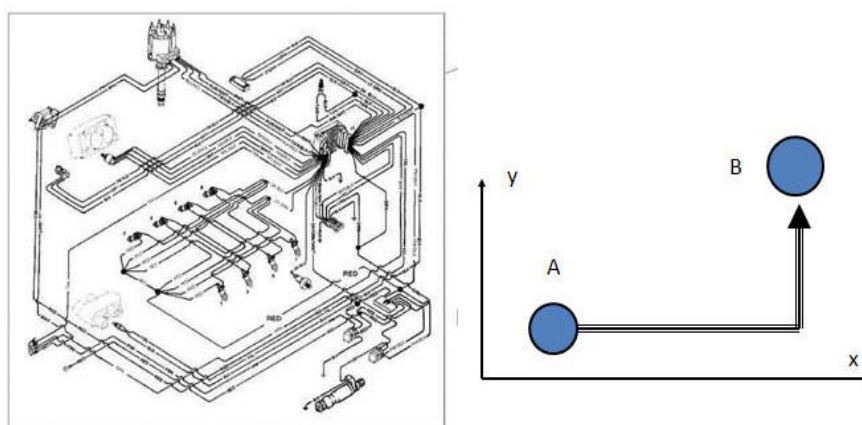


Рисунок 1.2 – Проектирование трассы с использованием манхэттенской метрики

Первоначально под каналовой поверхностью обычно понималась поверхность, являющаяся огибающей семейства сфер постоянного радиуса, центры которых лежат на плоской или пространственной кривой. Однако заданный таким образом канал относится лишь к узкому классу поверхностей каналового типа, а именно к трубчатым поверхностям. Позднее из гидравлики пришло несколько расширенное определение каналовой поверхности, как поверхности с нормальными конгруэнтными сечениями произвольной формы.

Известно более широкое геометрическое определение каналовой поверхности, по которому каналовой поверхностью названа кинематическая поверхность, образованная непрерывным каркасом замкнутых плоских поперечных сечений произвольной формы, не обладающих подобием, перпендикулярных касательным к пространственной кривой (оси канала), проходящей через центры тяжести их площадей, причем площади поперечных сечений должны удовлетворять некоторой наперед заданной зависимости.

Однако и это определение также не охватывает всего многообразия каналových поверхностей и законов их образования. В практике конструирования воздуховодов нашли широкое применение каналовые поверхности, образованные непрерывным каркасом замкнутых плоских поперечных контуров, лежащих во взаимно параллельных плоскостях (рисунок 1.3 а). Возможно построение каналových поверхностей, образованных перемещением вдоль осей образующих, плоскости которых наклонены под постоянным, не равным 90° , или переменным углом к касательным к осям каналов. Совсем не обязательным является условие прохождения расчетной линии тока (ось канала) через центры тяжести поперечных сечений.

Однако наиболее полно требованиям обеспечения заданных условий протекания потока газа или жидкости соответствует размещение плоскостей сечений перпендикулярно линии тока (рисунок 1.3 б). В этом случае с точки зрения геометрии каналовая поверхность определяется совокупность контуров поперечных сечений, плоскости которых перпендикулярны касательным к главной направляющей линии (оси) канала.

На наш взгляд, наиболее общее определение каналовой поверхности приведено проф. В.А.Осиповым [129]: “Каналовой назовём поверхность, образованную непрерывным каркасом замкнутых плоских определенным образом ориентированных в системе координат поверхности поперечных сечений, площади которых плавно изменяются по установленному закону”.

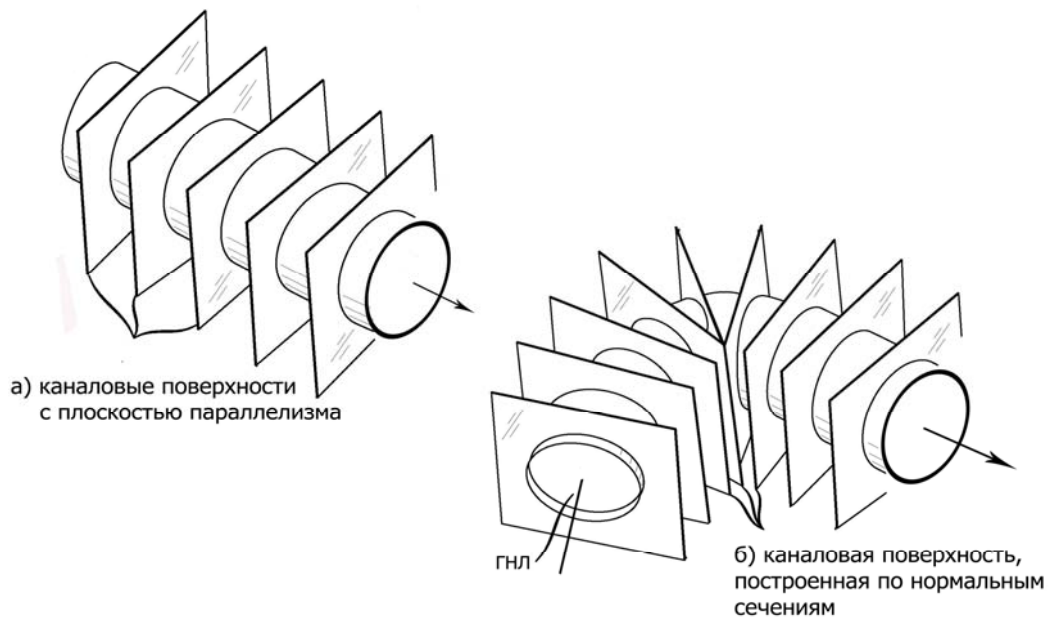


Рисунок 1.3 – Задание каналовой поверхности поперечными сечениями: а – сечения с плоскостью параллелизма; б нормальные сечения каналовой поверхности

Так как основное назначение каналовых поверхностей в технике заключается в переносе жидкостей или газов к различного рода силовым установкам и рабочим агрегатам, то в силу этого задача проектирования каналовой поверхности неразрывно связана с гидро- или аэродинамическими характеристиками потока. Основным элементом, связывающим все параметры сложной каналовой поверхности, является линия тока, называемая главной направляющей линией (ГНЛ). Это относится как к поверхностям с плоскостью параллелизма, так и к нормальным каналовым поверхностям. Она выявляется уже на стадии эскизных проработок изделия. Чаще всего в инженерной практике в качестве ГНЛ принята средняя линия тока (ось) каналовой поверхности. В общем случае это пространственная кривая. Задается она либо графически, либо аналитически, либо узловыми точками (дискретно). Пространственная ось каналовой поверхности в инженерной

практике обычно задается проекциями на координатные плоскости (рисунок 1.4). При компьютерном проектировании графическая или дискретная информация о ГНЛ должна быть преобразована и представлена в аналитической форме.

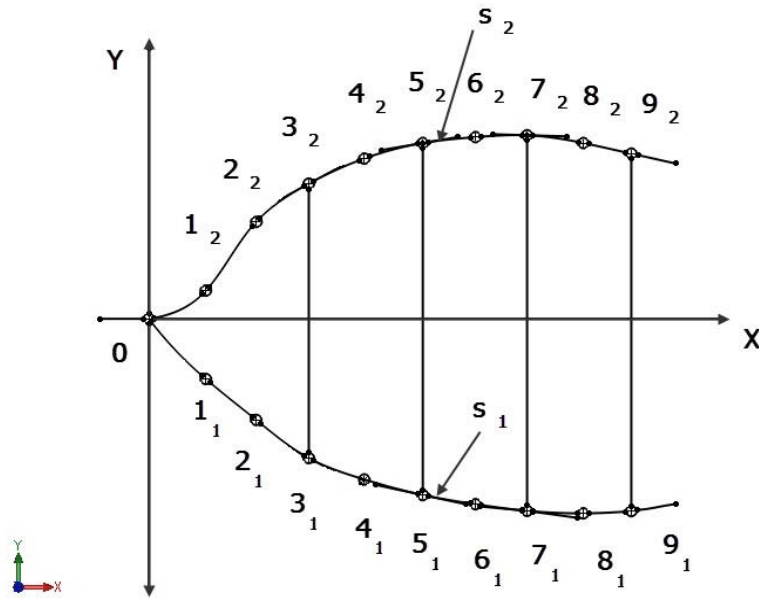


Рисунок 1.4 – Задание линии тока (ГНЛ) дискретным набором узловых точек

Количество и расположение узловых точек определяется требованиями порядка фиксации стыков участков пространственных кривых, образующих осевую линию канала. Так, например, нулевому порядку фиксации соответствует задание лишь координат x_i , y_i , z_i узловых точек. В этом случае без применения специальных приемов не обеспечивается единственность положения касательных на стыках. Первому порядку фиксации соответствует задание координат x_i , y_i , z_i узловых точек и направления касательных на стыках. В этом случае на каждом стыке имеем одну касательную и два основных триэдра (трехгранника Френе) из которых один соответствует предыдущему, другой — последующему участкам. Второму порядку фиксации соответствует задание координат узловых точек, касательных и

радиусов кривизны на стыках участков пространственной кривой. В этом случае на стыках совпадают и касательные, и основные триэдры.

Исходя из всего сказанного, можно констатировать, что подавляющее большинство исследований в этой области основано на представлении каналовой поверхности Φ как поверхности, образованной каркасно-кинематическим способом (рисунок 1.5). Управление параметрами как траектории (ГНЛ) s , так и формы сечения в перемещаемой плоскости Σ . В качестве метрических направляющих обычно берем контурные и характеристические линии. Все это позволяет сформировать поверхность с наперед заданными дифференциально-геометрическими характеристиками.

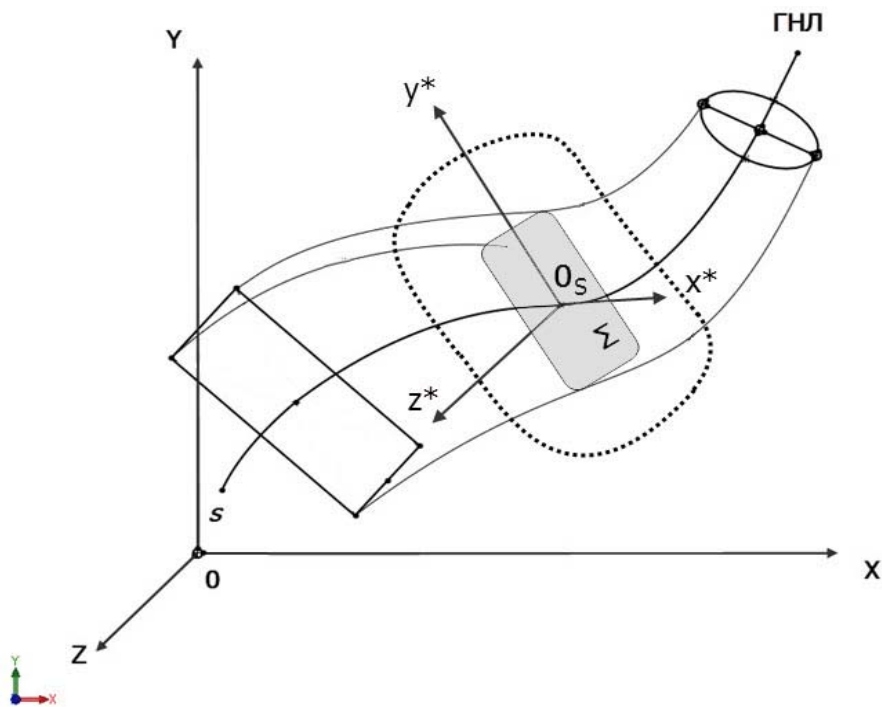


Рисунок 1.5 – Задание каналовой поверхности каркасно-кинематическим способом

Однако помимо дифференциально-геометрических условий на проектирование канала, не менее важным является выполнение заданных конструктором

технических условий, определяемых конкретным функциональным назначением, условиями компоновки, технологическими возможностями и т. д. Наиболее часто задаются следующие основные технические условия на проектирование: график площадей; форма характерных (узловых) поперечных сечений; продольные направляющие каналовой поверхности.

График площадей характеризует закон изменения площадей поперечных сечений вдоль принятой дистанции. За дистанцию обычно принимается прямолинейная ось исходной системы координат или криволинейная ось s проектируемого канала. Учитывая функциональное назначение канала, с точки зрения законов аэродинамики и газовой динамики более правильным является задание графика изменения площадей нормальных сечений вдоль оси s — средней линии тока рабочей жидкости. Однако ввиду простоты реализации, в качестве условия на проектирование часто задается график изменения площадей параллельных сечений вдоль прямолинейной оси $F(s)$, которые являются важнейшими техническими параметрами. Они входят во многие уравнения гидравлики, аэродинамики и газовой динамики, являясь связующим звеном между всеми параметрами, характеризующими поток рабочей жидкости.

Дополнительным важным условием проектирования каналовой поверхности является форма узловых поперечных сечений. Обычно исходя из конкретной конструкции изделия конструктором оговариваются формы начального и конечного сечений каналовой поверхности. Формы их определяются аэро- или гидродинамическими требованиями, компоновочными условиями и функциональным назначением каналов. Так, например, по аэродинамическим и конструктивным соображениям, начальные сечения воздухопроводов скоростных летательных аппаратов, как правило, прямоугольной формы, конечные сечения — круглой формы. Иногда оговаривается форма каких-либо промежуточных сечений.

Среди дополнительных технических характеристик каналовых поверхностей можно выделить: минимум гидравлических потерь в пограничном слое потока и

потоке в целом, повышение или понижение давления в потоке, поворот потока и др. В связи с этим необходимо проектировать каналовые поверхности так, чтобы они удовлетворяли наперед заданным техническим требованиям, а именно: гладкости обводов, описывающих поперечные и продольные образующие поверхности; равномерности кривизны направляющей линии; соблюдению определенной зависимости площадей поперечных сечений канала вдоль его длины.

Общим при проектировании любых каналовых поверхностей является требование обеспечения гладкости всех продольных направляющих. Это обусловлено стремлением получить высокие аэро- или гидродинамические качества каналов. Кроме перечисленных иногда могут выдвигаться и другие требования, определяемые особенностями конструкции или режимов работы. Например, условие плавного раздвоения на заданном участке, требование максимально возможного приближения формы поперечных сечений к кругу и т. д.

Базовая геометрическая модель каналовой поверхности (по монографии В.А.Осипова [132]) представлена на рисунке 1.6. На нем в декартовой прямоугольной системе координат $Oxyz$ задана пространственная кривая s — ось канала:

$$x_s = x_s(P); y_s = y_s(P); z_s = z_s(P).$$

По оси s как направляющей перемещается начало координат локальной системы $O_s x^* y^* z^*$. Угловое положение осей x^* , y^* , z^* задано функциями направляющих косинусов

$$\cos \alpha_1(P); \cos \beta_1(P); \cos \gamma_1(P);$$

$$\cos \alpha_2(P); \cos \beta_2(P); \cos \gamma_2(P);$$

$$\cos \alpha_3(P); \cos \beta_3(P); \cos \gamma_3(P);$$

или функциями углов Эйлера (для ортогональной системы) $\varphi = \varphi(P); \psi = \psi(P); \theta = \theta(P)$.

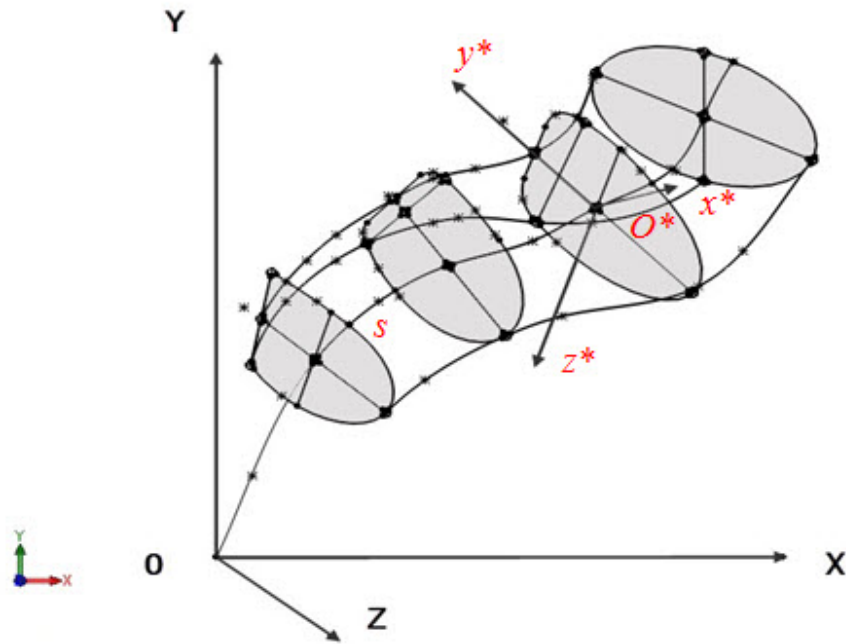


Рисунок 1.6 – Геометрическая иллюстрация базовой геометрической модели каналовой поверхности (на основе [132])

В последнем случае направляющие косинусы определяются по углам Эйлера:

$$\cos \alpha_1 = \cos \varphi \cos \psi - \sin \varphi \sin \psi \cos \theta ;$$

$$\cos \beta_1 = \sin \varphi \cos \psi + \cos \varphi \sin \psi \cos \theta ;$$

$$\cos \gamma_1 = \sin \varphi \sin \theta ;$$

$$\cos \alpha_2 = -\cos \varphi \cos \psi - \sin \varphi \sin \psi \cos \theta ;$$

$$\cos \beta_2 = -\sin \varphi \sin \psi + \cos \varphi \cos \psi \cos \theta ;$$

$$\cos \gamma_2 = \cos \psi \sin \theta ;$$

$$\cos \alpha_3 = \sin \varphi \sin \theta ;$$

$$\cos \beta_3 = -\cos \varphi \sin \theta ;$$

$$\cos \gamma_3 = \cos \theta ;$$

В плоской системе координат $O_s y^* z^*$ задан переменный образующий контур $y^* = y^*(z^*, P)$. Записав образующий контур в исходной системе координат $Oxyz$

$$x = y^* \cos \alpha_2 + z^* \cos \alpha_3 + xO_s ;$$

$$y = y^* \cos \beta_2 + z^* \cos \beta_3 + y_{0s};$$

$$z = y^* \cos \gamma_2 + z^* \cos \gamma_3 + z_{0s};$$

и придавая величинам переменные значения, получим систему уравнений

$$x = y^* (z^*, P) \cos \alpha_2 (P) + z^* \cos \alpha_3 (P) + x_s (P);$$

$$y = y^* (z^*, P) \cos \beta_2 (P) + z^* \cos \beta_3 (P) + y_s (P);$$

$$z = y^* (z^*, P) \cos \gamma_2 (P) + z^* \cos \gamma_3 (P) + z_s (P).$$

Эта система уравнений и является обобщенной математической моделью кинематической каналовой поверхности. По существу это параметрическое уравнение поперечного каркаса кинематической каналовой поверхности общего вида.

По геометрическому и математическому моделированию каналовых поверхностей выполнено достаточное число исследований специалистами в области прикладной геометрии, представителями различных отраслей промышленности. В первую очередь следует отметить работы, выполненные профессором Осиповым В.А. [129, 130] и его многочисленными учениками - Андреевым В.А. [5], Зелевым В.П. [54], Лелюшенко С.И. [86], Мезенцевым Л.Г. [99], Поликарповым Ю.В. [140], Миролубовой Т.И. [101, 102], Давыдовым Ю.В. [38], Егоровым Э.В. [47, 48] а так же другими исследователями московской и общероссийской геометрической школы - Александровичем В.П. [3], Дёминой В.А. [41], Ивановым Г.С. [61, 62, 63], Кулишом А.С. [82], Хасановым В.Х. [197], Найдышем В.М. [111, 112, 113] и др. Следует отметить большой вклад в решение этого вопроса Киевской научной школы прикладной геометрии - труды Блюка А.В. [16], Василевского О.В. [20], Грибова С.М. [35], Дорошенко Ю.А. [42, 43], Кожушко Н.А. [71], Пилипаки С.Ф. [139], Радзивилловичем В.В. [149], Обуховой В.С. [126, 127], Скидана И.А. [173] и др.

1.2 Основные виды геометрических моделей компоуемых объектов

Теоретические вопросы метода геометрического моделирования с различных аспектов исследовались в трудах отечественных ученых Валькова К.И. [19], Денискина Ю.И. [39, 40], Иванова Г.С. [63, 64, 65], Котова И.И. [77, 78], Михайленко В.Е. [103, 104, 105], Осипова В.А. [131, 132], Полозова В.С. [80, 141], Похвалинского А.Б. [144, 145], Рвачева В.Л. [150, 151], Рыжова Н.Н. [158, 159, 160], Стародетко Е.А. [177, 178, 179, 180], Фролова С.А. [196], Якунина В.И. [204, 205, 206] и др, а также зарубежных Поля де Кательджо, Роберта Фергюсона (Robert Fergusson), Стивена Кунса (Steven Coons), Пьера Безье (Pierre Bézier), Шарля Эрмитта (Charles Hermite), Исаака Шенберга (Isaac Jacob Schoenberg), Карла де Бура (Carl de Boor), Кена Версприлла (Ken Versprille), Юджина Ли (Eugene Lee), Стива Гейзберга (Steve Ginsberg) и других. В их трудах содержатся как классические, так и компьютерно - ориентированные методы задания, расчета и воспроизведения геометрических форм.

Разумеется, такое обилие методов ориентировано на описание геометрической формы разнородных технических объектов. Нам представляется наиболее рациональных для решения технических задач классификатор геометрических объектов, предложенный Семенковым О.И. – Осиповым В.А. [165, 166, 132, 164, 165], в основе которого лежит структура синтеза геометрических объектов из составляющих их элементов. Этот классификатор подразделяет все геометрические объекты (ГО) на две большие группы – геометрические объекты сложной технической формы и геометрические объекты сложной технической структуры. Объекты первой группы ограничиваются отсеками поверхностей, каждая из которых описана достаточно сложными аналитическими уравнениями или системами уравнений. К ним относятся фюзеляжи самолетов, кузова автомобилей, корпуса судов, лопасти турбин и т.п. Объекты второй группы представляют собой объеди-

ненные на основе теоретико-множественных операций (объединение, пересечение, отрицание) ГО, как правило, относительно простых геометрических форм. В трудах Маркина Л.В. по геометрическим моделям компоновки исходя из специфики решаемых задач количество таких групп увеличено до четырех (рисунок 1.7) [2, 74].

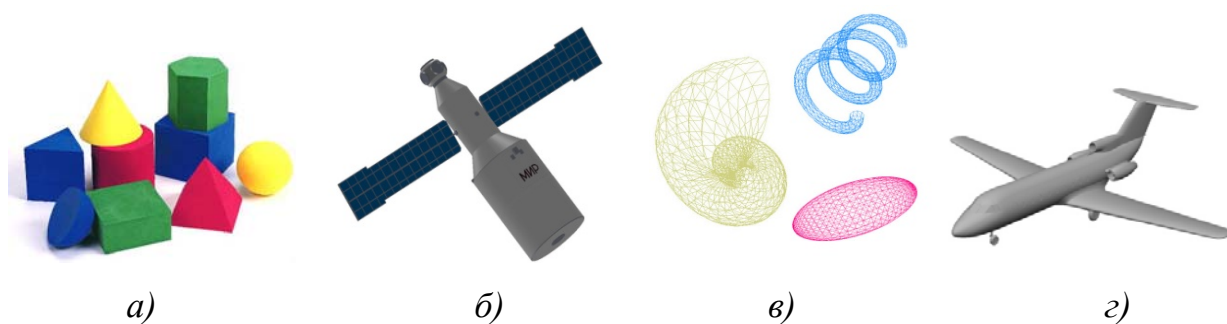


Рисунок 1.7 – Классификация геометрических объектов по уровню сложности: а – примитивы; б – объекты сложных технических структур; в – объекты сложных технических форм; г – объекты сложных технических форм и структур

Обилие геометрических форм объектов в технике, строительстве, дизайне и т.п. требует библиотеки методов геометрического моделирования, наиболее адаптированных для описания конкретных особенностей формы геометрических объектов. Поэтому помимо классификации геометрических объектов существуют и системы классификации геометрических моделей. Одним из таких классификационных признаков, связанных с информационной структурой геометрической модели, является степень информационной насыщенности объекта. Известна классификация ГО по информационной насыщенности (рисунок 1.8). Она отражает точность описания геометрического объекта (только по ребрам, по сечениям, по всей поверхности, по всему телу). Очевидно, что любое увеличение точности описания (информационной насыщенности) приводит к увеличению сложности математического описания и его компьютерной реализации.

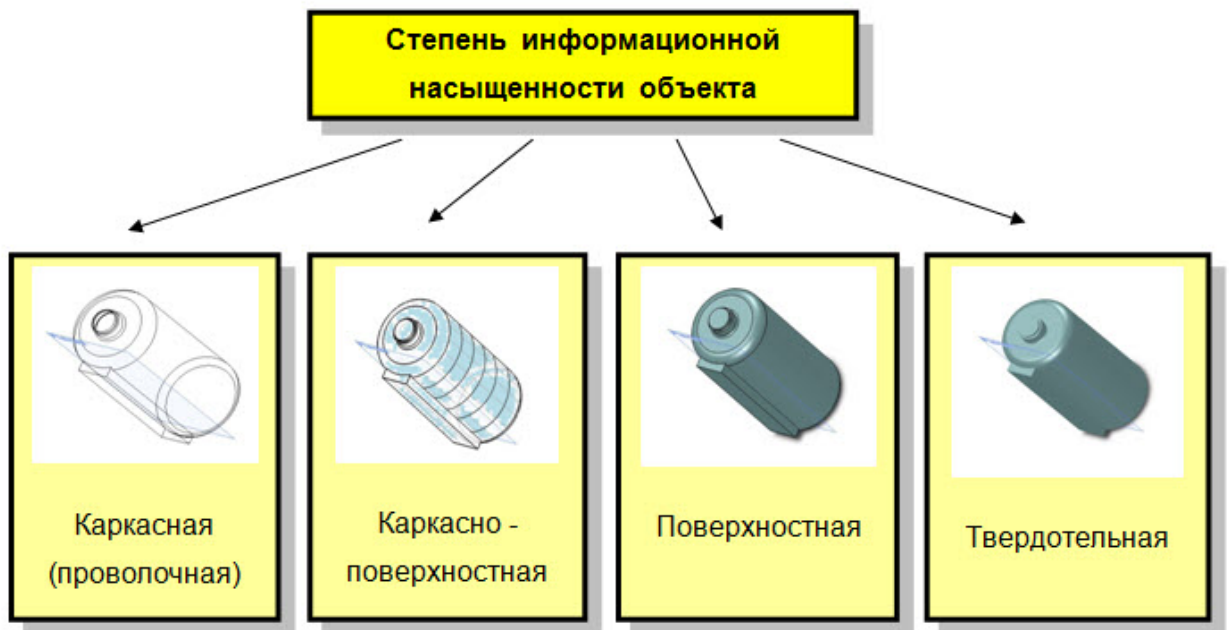


Рисунок 1.8 – Классификация геометрических моделей по информационной насыщенности

Классификацию же самих методов геометрического моделирования можно разделить на три класса (рисунок 1.9).

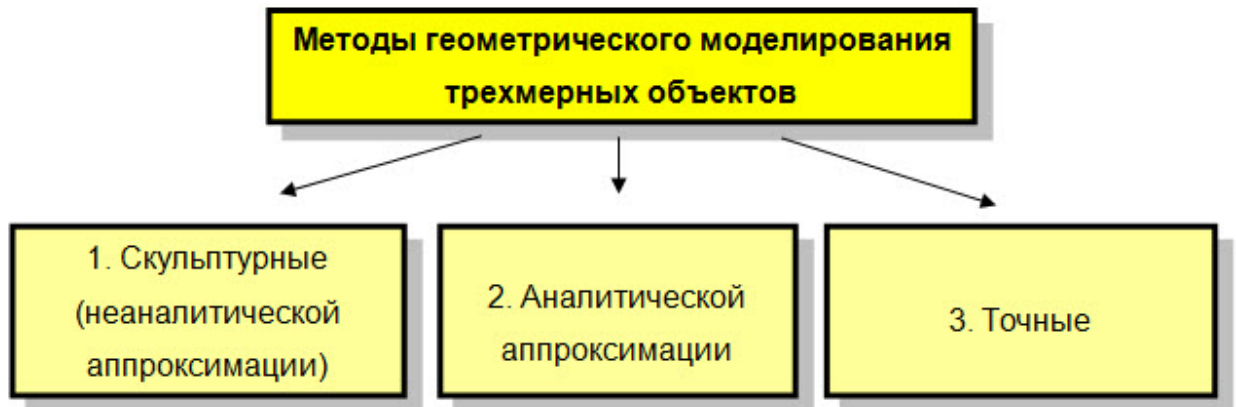


Рисунок 1.9 – Классификация методов моделирования геометрических объектов

Скульптурные методы используются для создания геометрических моделей таких объектов, точное аналитическое описание которых неизвестно и вряд ли может быть получено. Не случайно они называются “скульптурными” – в своей

работе не думая о математике и геометрии скульптор просто удаляет из камня “все лишнее” (рисунок 1.10). Помимо дизайна скульптурные методы широко применяются и в технике (авиа-, судо- и автостроении), когда форма поверхности корректируется не только по эстетическим соображениям, но и на основании данных аэродинамического или гидродинамического эксперимента (рисунок 1.11). Однако в итоге мы получаем аналитическое выражение геометрической формы этих объектов с той или иной степенью точности.

Реализация этого метода основана на довольно многочисленной библиотеке методов аппроксимации поверхностей с использованием сплайнов, В-сплайнов, NURBS, поверхностей Кунса, Эрмита, Лагранжа, Безье и т.п.



а)



б)

Рисунок 1.10 – Пример скульптурной поверхности (а) и работа скульптора над ее созданием (б)



Рисунок 1.11 – Примеры технических объектов, реализованных скульптурными методами

Методы аналитической аппроксимации используются для описания формы объектов, состоящих из сложных поверхностей второго и более высоких порядков. Поскольку прямая вычислительная обработка поверхностей столь сложной геометрической формы затруднена, они аппроксимируются участками поверхностей более низкого порядка (плоскостями, цилиндрами, сферами и т.п.) – рисунок 1.12.

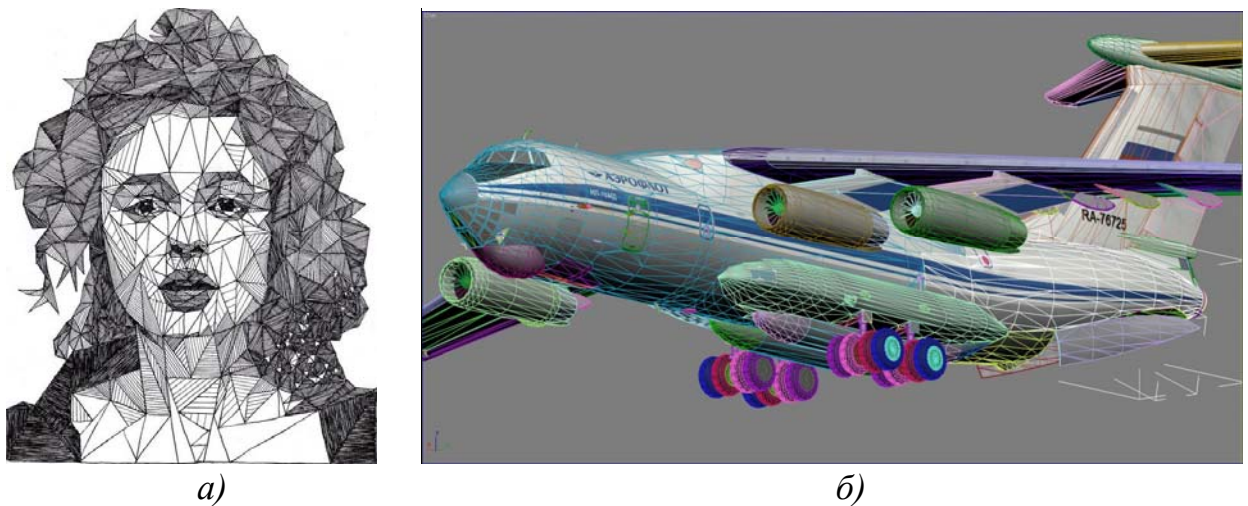


Рисунок 1.12 – Пример объектов очень сложной геометрической формы, полученной методом аналитической аппроксимации: а – биологического; б - технического

Точные методы моделирования трехмерных объектов представляют собой совокупность следующих известных методов:

- Кинематического;
- Параметрического;
- Проволочно-каркасного;
- Кусочно - аналитического;
- Алгебро-логического (метод R-функций);
- Метод “разложения на элементы”;
- Метод конструктивной геометрии элементарных объемов.

Иллюстрация этих методов геометрического моделирования приведена на рисунке 1.13.

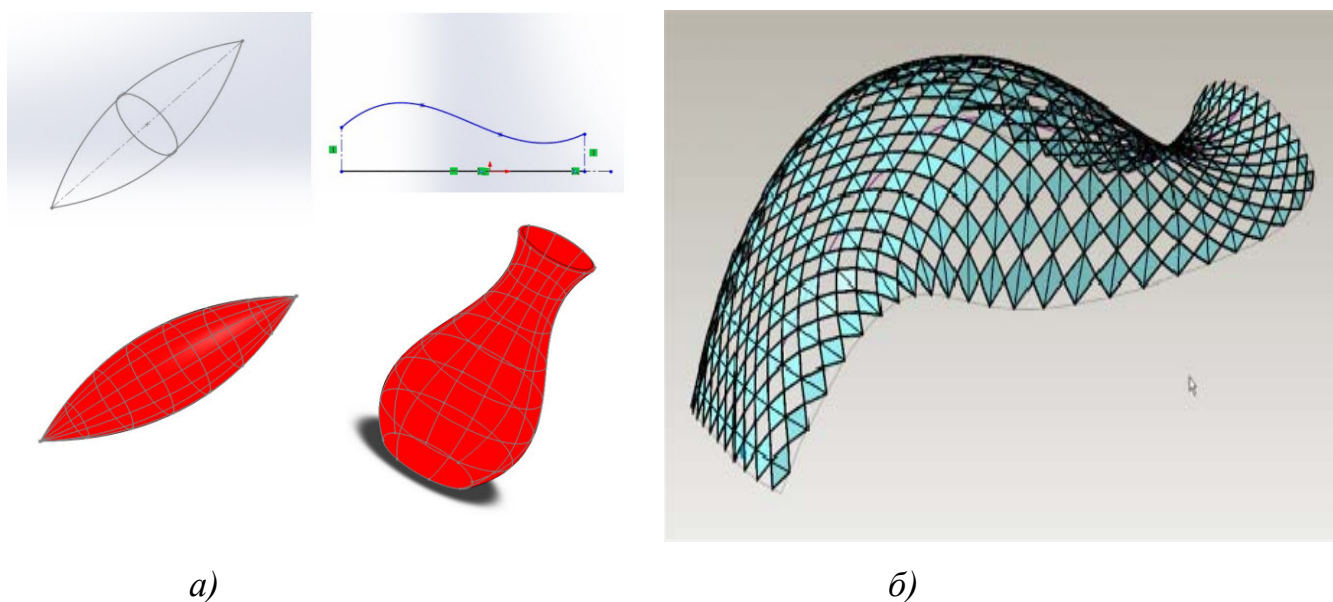


Рисунок 1.13 – Иллюстрация точных методов моделирования трехмерных объектов: *a* – кинематического; *б* – параметрического и т.д.

1.3 Основные геометрические модели размещения компонуемых объектов в пространстве

Качество компоновки любого технического изделия во многом определяет его техническое совершенство и эксплуатационные характеристики. Особенно эта проблема актуальна для транспортного машиностроения, где увеличение габаритных размеров вызывает дополнительное сопротивление окружающей среды при движении транспортного средства. И исключительно актуальна эта проблема для авиационной и ракетно-космической техники с ее высокими скоростями полета, сложными геометрическими формами и высокой плотностью компоновки.

На заре авиации компоновка осуществлялась на основе опыта изготовления и эксплуатации предыдущих образцов авиационной техники, однако внедрение чертежного способа проектирования позволило производить поиски приемлемых проектных решений на чертежах и тем самым отделить процесс проектирования самолетов от его производства [5, 9]. Однако при высокой плотности компоновки современной авиакосмической техники, даже самое тщательное выполнение чертежей компонуемых объектов не исключает возможность случаев взаимного пересечения компонуемых объектов. Эти коллизии проектирования позволяют устранить физические макеты компоновок в натуральную величину, выполненные из легкообрабатываемых материалов (дерево, фанера, пенопласт, легкие сплавы), но их изготовление значительно удлинит и удорожает проектирование авиационно-космической техники.

Принципиально новые возможности компоновки появились с использованием компьютерных методов твердотельного моделирования, позволяющих не только создавать виртуальные модели компоновок, но и с высокой точностью проверять на них возможные случаи взаимного пересечения компонуемых объектов (рисунок 1.14). Однако даже эти совершенные методы проектирования и компьютерного моделирования анализируют лишь существующую конструкцию, элементы которой получены проектантом с использованием небольшого набора типовых операций (выдавливание, выдавливание по траектории, вращение, моделирование по сечениям, деформации), что не позволяет моделировать наиболее сложные “скульптурные” объекты, а для объектов более простых форм не гарантирует оптимальность конструкции. Таким образом виртуальная модель технического объекта, созданная в любой системе геометрического моделирования (СГМ) – это не более чем реализованная средствами СГМ с учетом личного опыта проектанта конкретная проектная разработка технического объекта (не факт, что наилучшая).

Проектирование современной высокотехнологичной техники невозможно без использования средств автоматизированного проектирования, компонентом кото-

рой являются средства компьютерной графики. Основными трудами в этой области мы считаем труды отечественных ученых Горелика А.Г. [31, 32, 33, 34], Норенкова И.П. [115, 116, 117], Сидоренко С.М. [168], Вермишева Ю.Х. [21, 22], Петренко А.И. [137], Советова Б. Я. [176], Прохорова А.Ф. [146], Курейчика В.М. [84], Ли К. [85], а также ряда зарубежных авторов - Гардана И. (Yvon Gardan) и Люка М. (Michel Lucas) [26], Грувера М. (Mikell P. Groover) и Зиммерса Э. (Emory W. Zimmers) [37], Гиллоя В. (Wolfgang K. Giloi) [30], Принса М. [147], Шенена П. (Peter Shenen) [97], Шпура Г. (Gunter Spur) и Краузе Ф. (Frank-Lothar Krause) [203], Энгельке У. (William D. Engelke) [204], Хокса Б. (Barry Hawkes) [198], Хорафаса Д., (Dimitris N. Chorafas) и Легга С. (Stephen J. Legg) [199] и других.

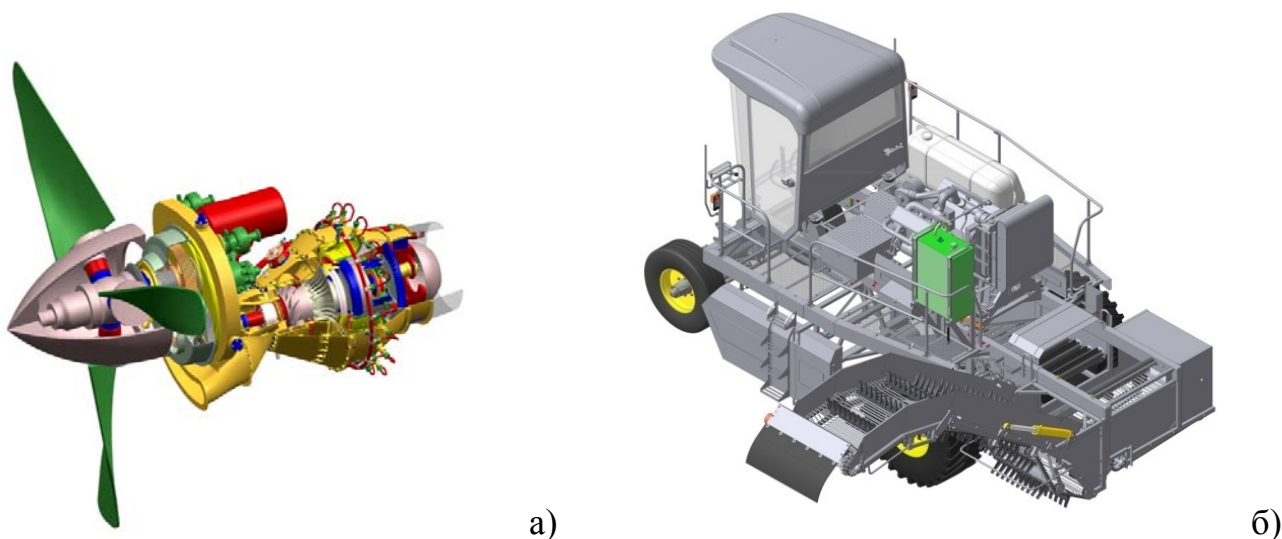


Рисунок 1.14 – Пример компьютерного твердотельного моделирования конструкции турбовинтового двигателя (а) картофелеуборочного комбайна (б)

Однако наибольшее развитие теории и практики автоматизированного проектирования было уделено в авиакосмической отрасли как наиболее высокотехнологичной области современной техники. Следует отметить значительный вклад в развитие автоматизированного проектирования авиационно- космической техники следующих отечественных ученых – Волошина В.В. [24], Осина М.И. [128, 106], Формалева В.Ф. [195], Мальчевского В.В. [94], Лисейцева Н.К и Самойловича

О.С. [46], Куприкова М.Ю. [2, 83, 40], Падалко С.Н. [135, 178], Пухова А.А. [2, 148] и др.

Как уже отмечалось выше, современные методы геометрического моделирования позволяют аналитически описать геометрические формы практически любой степени сложности, однако это не приближает нас к проблеме компьютерной автоматизированной компоновки, для которой важнее не точность описания, а другие специфические свойства геометрической модели:

- Возможность относительно просто определять случаи взаимного пересечения скомпонованных объектов;
- Возможность на основе данной модели генерировать алгоритмы рационального размещения объекта в пространстве.

Вместе с тем известны исследования, в которых вопросы компоновки даже таких геометрически сложных объектов, как каналовые поверхности, решались традиционными методами геометрического моделирования каналовых поверхностей с учетом дополнительных функциональных или компоновочных требований.

Чаще всего это проектирование трасс простейшей геометрической формы между уже скомпонованными объектами (например, химическими аппаратами или выпускными каналами двигателя). Это исследования Образцова А.А. [125], Брысина В.А. [18], Богацкого И.З. [17], Драганова Б.Х. [44, 45], Искакова С.Д. [67], Кожушко Н.А. [71], Мартыновой О.Г. [98], Некрасовой О.И. [114] и других. Однако все эти исследования являются лишь частными решениями геометрических моделей каналовых поверхностей.

Типичными представителями этого направления являются исследования Саккиевой М.К. [161, 162], Безкоровайного В.П. [14, 15], Есмухановой Ж.Ж [49, 50, 51], Попова Ю.И. [143] и др.

Все описанные выше методы геометрического моделирования каналовых поверхностей ориентированы на то, что каналовая поверхность проектируется как бы “сама по себе” и не способна оперативно изменять свою геометрическую фор-

му при изменении ее положения среди других уже скомпонованных объектов. Система трубопроводов представляется графом, в вершинах которого пункты сбора (скважины, емкости, химические аппараты и т.п.), а соединения между ними – ветвями моделируемого графа. Целью проектирования является минимизация графа, связывающего заданные вершины с учетом некоторых дополнительных факторов (например, реальной геометрии поверхности, на которой прокладываются трубопроводы). Пример такого объекта с многочисленными соединительными трассами приведен на рисунке 1.15.



а)



б)

Рисунок 1.15 – Примеры технических объектов с многочисленными коммуникациями: а – химические аппараты, между которыми проектируются соединительные трассы; б – пример компоновки соединительных трасс

Необходимым, хотя и недостаточным условием получения любой компоновки является реализация условия взаимного непересечения компонуемых объектов (УВН). При этом оболочка, в которой происходит сама компоновка, также рассматривается как компонуемый объект, с которым также не должно быть взаимного непересечения. Помимо этого не должно быть взаимного пересечения с объектами, которые являются областями запрета. Если эти условия соблюдаются,

то мы получаем всего лишь одну из возможных компоновок, а их вариантов (в зависимости от порядка размещения компонуемых объектов. Методы и геометрические модели (в том числе эвристические), учитывающие специфику именно компоновочных задач (как для 2D, так и для 3D случаев), были разработаны в фундаментальных трудах отечественных ученых проф. Гаврилова В.Н. [25], проф. Мухачевой Э.А. [107], акад. Рвачева В.Л. [150, 151], проф. Стояна Ю.Г. [184, 182, 183], проф. Маркина Л.В. [95] и других, а также зарубежных ученых Bortfeldt A.[213], Cagan J. [215], George J.A.[222], Gilmore P.C.[223], Lim A.[230], Lodi A.[231], Martello S.[233], Pisinger D.[241], Robinson D. F.[247], Saaty T. L. [249], Szykman S. [252], Vigo D. [254] И др.

Анализ научной литературы в области автоматизации компоновки показал, что несмотря на значительное количество исследований в этой области и возможность с использованием современных информационных технологий аналитически описать поверхность с геометрической формой практически любой сложности, решены лишь отдельные, частные задачи автоматизированной компоновки. Это либо проверка условия непересечения в уже сгенерированной проектантом компоновки внутренними средствами системы геометрического моделирования (СГМ) одной из систем твердотельного моделирования (КОМПАС, Solid Works, Catia и др.). Но в этом случае средства компьютерной графики лишь в качестве инструментального средства уже готового технического решения, основанного лишь на опыте и интуиции проектанта.

Разработанные на настоящий момент методы и алгоритмы компоновки, а также реализующие их программные средства ориентированы на проверку условия непересечения лишь объектов простейших геометрических форм (примитивов и композиций примитивов). Для этого используются библиотека специальных методов определения условия непересечения (УВН) – метод аппроксимирующих сфер, метод разделяющей плоскости, использование функции плотного размещения (Ф-функции), R-функций, метод минимального зазора и ряд других методов,

достаточно подробно описанных в [53]. Алгоритмы последующего размещения компоуемых объектов основаны либо на переборных методах, либо на эвристических алгоритмах, но после такого автоматического размещения проверка условия непересечения возможна, как уже отмечалось, лишь для объектов несложных геометрических форм (примитивов и композиций примитивов). Число возможных вариантов размещения даже сравнительно небольшого количества компоуемых объектов выражается астрономическими числами, поэтому сам процесс автоматизированной компоновки сводится к генерации случайной комбинации размещения, определения параметров эффективности данного варианта компоновки и запоминания рекордных значений определенных вариантов компоновки. Если новый вариант компоновки окажется по параметру эффективности лучше, чем предыдущий рекорд, то он запоминается со своим новым значением параметра эффективности, которое становится новым рекордным значением эффективности. Если же он окажется хуже, то этот вариант компоновки просто забывается и генерируется следующий. Учитывая многовариантность комбинаций компоновки, таким методом практически невозможно выйти на глобальный оптимум эффективности, но возможно получение рациональных вариантов компоновки, удовлетворяющих заданным техническим требованиям. Необходимо отметить, что постоянное увеличение производительности компьютерной техники позволяет все более приближаться от рациональных вариантов компоновки к оптимальным.

Таким образом, автоматизированная компоновка изделий сложных технических форм (типа каналовых поверхностей) представляет собой сложную и в настоящее время не решенную техническую задачу. Разработка методов и алгоритмов автоматизированной компоновки таких объектов представляет собой предмет исследования настоящей диссертации.

ЗАКЛЮЧЕНИЕ И ВЫВОДЫ К ГЛАВЕ 1

Сложность современной транспортной техники, особенно авиакосмической, с высокой плотностью компоновки, значительно усложняет процесс ее проектирования. Проведенный анализ научной литературы позволяет сделать следующие выводы:

1. Компоновка современной высокотехнологичной техники (особенно транспортной) во много определяет ее эксплуатационно-технические свойства. Особенно актуально обеспечение качества компоновки для авиационно-космической техники, отличающейся высокой плотностью компоновки.

2. Применение современных методов геометрического моделирования и информационных технологий позволяет аналитически описать объекты достаточно сложных геометрических форм, к которым относятся каналовые поверхности.

3. Современные системы твердотельного моделирования позволяют своими инструментальными средствами с учетом опыта и интуиции проектанта (без оптимизации самого процесса компоновки) формировать виртуальные модели компоновок и определять в них условия взаимного непересечения.

3. Проведены достаточно обширные исследования в области разработки методов и алгоритмов автоматизированной компоновки позволили разработать библиотеку методов проверки условия взаимного непересечения, которые однако применимы лишь к объектам простых технических форм (примитивов и композиций примитивов).

4. Разработаны алгоритмы и геометрические модели, которые позволяют рационально (а в некоторых случаях и оптимально) размещать плоские (2D) объекты сложной формы, либо пространственные (3D) объекты простых форм на основании либо переборных, либо эвристических алгоритмов с последовательным приближением к рациональным решениям.

5. Вопросы автоматизированной компоновки объектов сложных технических форм (типа каналовых поверхностей) в настоящее время решены лишь для некоторых частных случаев. Нахождение общих решений автоматизированной компоновки таких объектов является актуальной технической задачей.

2 РАЗРАБОТКА И ИССЛЕДОВАНИЕ РЕЦЕПТОРНЫХ ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ В ЗАДАЧАХ ТЕЛЕСНОЙ ТРАССИРОВКИ

2.1 Физическая постановка задачи исследования

Физическая постановка задачи исследования заключается в следующем – необходимо провести ряд соединительных трасс в заданных точках входа и выхода между уже размещенными объектами. Пример такой ситуации для двигателя внутреннего сгорания автомобиля приведен на рисунке 2.1, на котором пронумерованы сами трассы и точки их входа и фиксации.

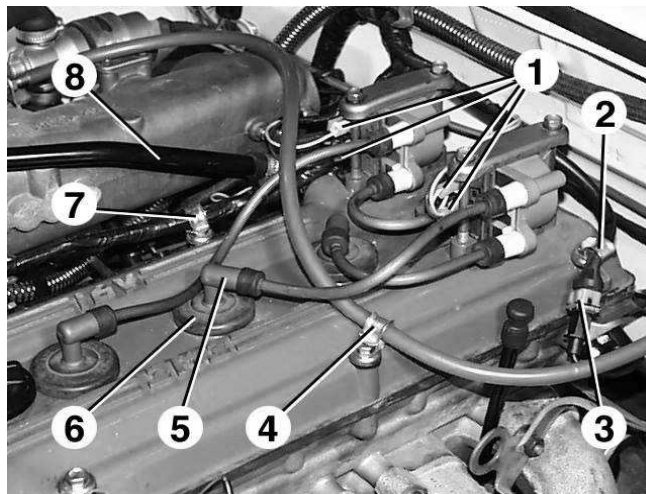


Рисунок 2.1 – Проведение набора трасс в техническом устройстве

Необходимо отметить, что изображенный на рисунке 2.1 случай является простейшим примером проектирования трассы, т.к. в данном примере:

- имеется сравнительно небольшое количество трасс, которые необходимо спроектировать;
- все трассы одного сечения по длине самой трассы;
- в данном примере невысокая плотность компоновки;

- большинство трасс являются электрическими жгутами, и следовательно, допускают при своем проектировании любые углы изгибов.

Проиллюстрируем сначала постановку задачи телесной трассировки для простоты в 2D (плоской) постановке и без учета размеров самой трассы (т.е. при отсутствии “телесности”) и заданных ограничений на ее плавность.

Имеется прямоугольная область размерами $X \times Y$, в которой расположены области запрета. Задана точка входа A и точка выхода B канала (рисунок 2.2). Необходимо провести трассу между заданными начальной и конечной точками A и B .

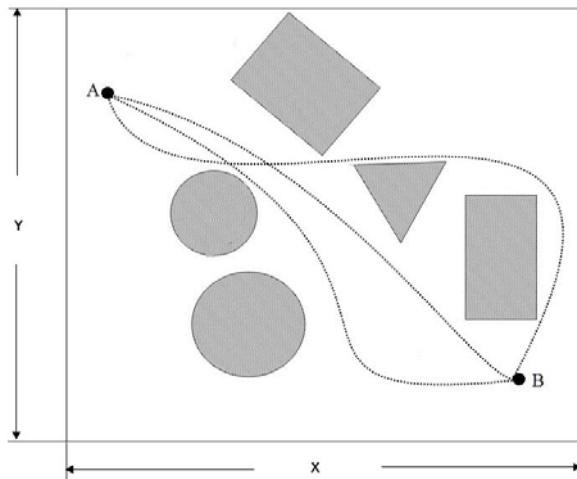


Рисунок 2.2 - Нахождение рационального пути между двумя конечными точками A и B в 2D постановке без учета размеров самой трассы и ограничений на плавность

Из рисунка 2.2 видно, что поскольку мы не наложили никаких ограничений на трассу, возможно достаточно много вариантов ее прохождения. Учитывая нашу транспортную специфику проектирования техники, очевидно, что из всех трасс на рисунке 2.2 лучше будет та, которая короче. Однако из практики проектирования каналов известно [96, 53, 66, 202], что при прохождении жидкости или газа по каналу неизбежны *гидравлические потери* или *гидравлическое сопротивление* — безвозвратные потери удельной энергии (переход её в теплоту) на

участках гидравлических систем (в том числе и трубопроводах), обусловленные наличием вязкого трения. Гидравлические потери принято разделять на два вида:

- **потери на трение по длине** — возникают при равномерном течении, в чистом виде — в прямых трубах постоянного сечения, они пропорциональны длине трубы;
- **местные гидравлические потери** — обусловлены т. н. *местными гидравлическими сопротивлениями* — прежде всего изменениями формы и размера канала, деформирующими поток.

Для уменьшения гидравлических потерь мы будем вынуждены наложить на прохождение трассы дополнительное требование плавности, то самая короткая трасса прохождения пути может не удовлетворять этому критерию, поэтому нам возможно придется искать трассу более длинную, но зато плавную. Если же таких возможных трасс окажется несколько, то выберем из них самую короткую.

Усложним задачу проектирования. Будем считать, что наша трасса должны быть не бесконечно тонкой, как на рисунке 2.3, и иметь вполне конкретные физические размеры, т.е. являться каналовым объектом. В простейшем случае это цилиндрический канал постоянного радиуса R (рисунок 2.3 а), в более сложном — канал переменного сечения (рисунок 2.3 б).

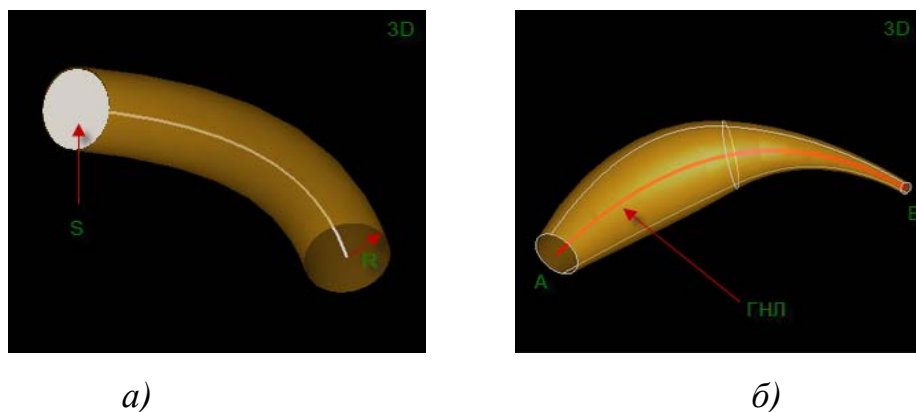


Рисунок 2.3 – Варианты каналов: а – постоянного сечения; б – переменного сечения

Известно, что положение канала в пространстве определяется, прежде всего, положением его линии тока, которая в геометрии называется главной направляющей линией (ГНЛ). Если канал постоянного сечения, как на рисунке 2.3 *а*, для его однозначного геометрического описания достаточно задания описания ГНЛ и значения радиуса R . Если же он переменного сечения, как на рисунке 2.3 *б*, то необходима дополнительная информация о форме канала. Например, если он представляет собой в сечении круг переменного радиуса (рисунок 2.4 *а*), то это может быть закон изменения площади по длине ГНЛ (рисунок 2.4 *б*).

Разумеется, в практике проектирования авиационной техники встречаются и каналы более сложной геометрической формы, но мы остановимся на этом уровне сложности – в нашем исследовании наиболее сложный случай – канал с пространственной ГНЛ переменной площади сечения круглой формы.

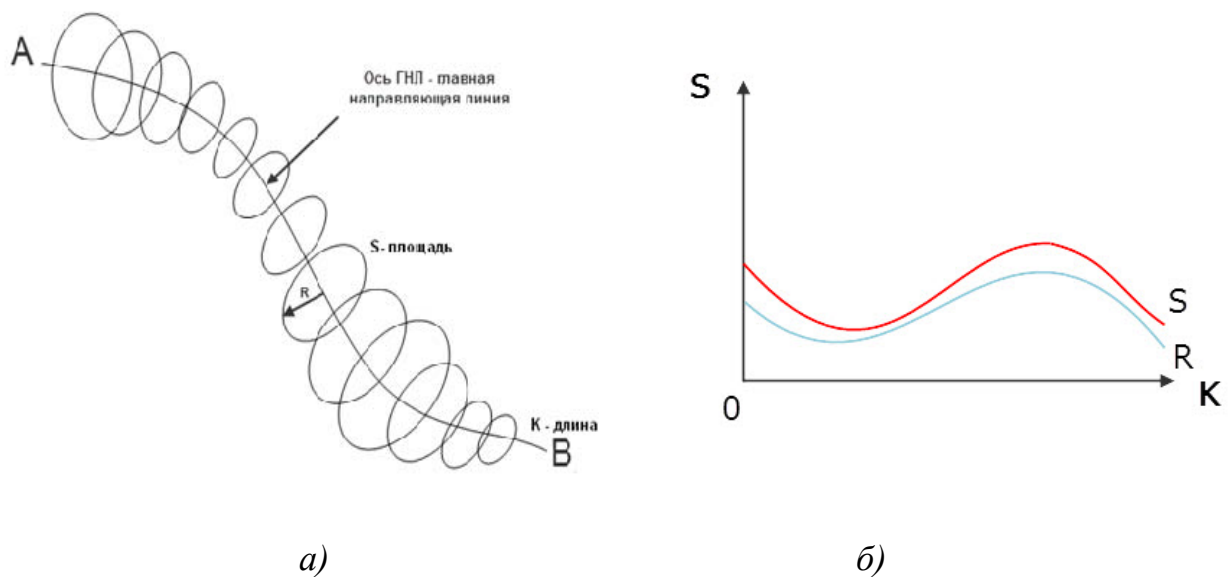


Рисунок 2.4 – Задание формы канала переменного сечения: а – вид сечений по длине ГНЛ; б – задание функции площади сечения S и радиус круга R по длине ГНЛ

Однако в нашем случае будет действовать еще одно ограничение - кривизна ГНЛ не должна быть меньше заданной. Очевидно, что ГНЛ, являясь кривой лини-

ей, в каждой точке имеет свой радиус кривизны R' (рисунок 2.5). Нужно, чтобы его значение было не меньше заданного конструктором значения R_{min} . Выполнение это условия необходимо для исключения ситуаций резкого изгиба трубопровода, что приводит к срыву потока, увеличению его гидравлического сопротивления, забивания трубопровода грязью и т.п..

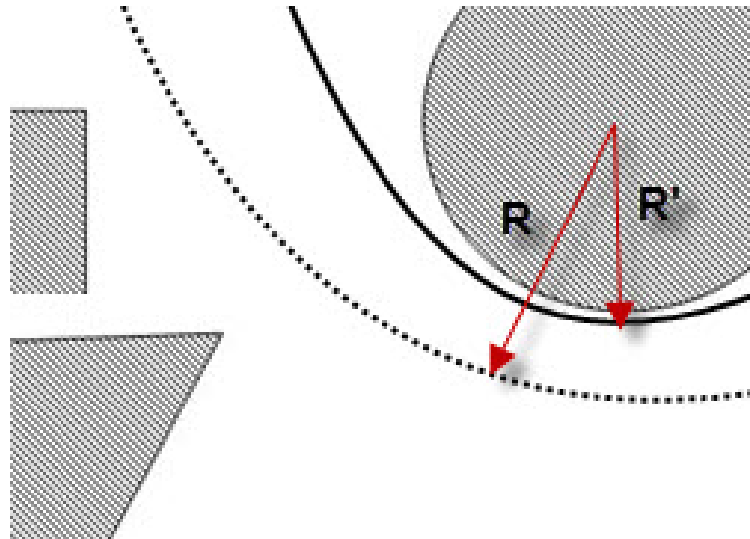


Рисунок 2.5 – Требование обеспечения минимального радиуса кривизны канала (в 2D постановке)

Еще одним дополнительным требованием к проектированию каналовой поверхности может быть не только обеспечение условия взаимного непересечения с уже скомпонованными объектами, но и обеспечения заданного минимального расстояния между ними, например для обеспечения требуемой тепло- или электроизоляции (рисунок 2.6). Будем называть это расстояние δ –окрестностью, который задается как условие проектирования – это минимальное расстояние от края трубопровода до размещенного объекта

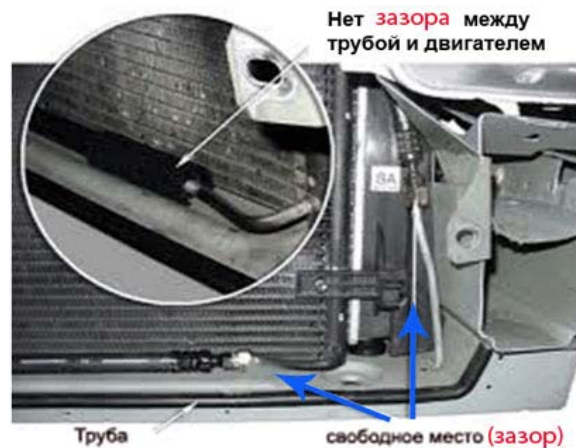


Рисунок 2.6 – Необходимость обеспечения требуемого зазора δ при проектировании канала

При проектировании электрических соединительных трасс к прохождению жгутов не предъявляются такие жесткие требования по плавности, как для пнево-гидротрасс (хотя сохраняются технологические требования обеспечения минимального радиуса сгиба жгута при монтаже), однако возможны случаи взаимного влияния электрических, электромагнитных или тепловых полей как на соседние жгуты, так и на отдельные уже скомпонованные объекты (как правило, приборное оборудование). Компоновка трасс с учетом такого взаимовлияния является предметом специального исследования в данной диссертации не рассматривается. Подытожим допущения физического характера, которые будут приняты в настоящем исследовании:

1. Размеры трасс сопоставимы с размерами компонуемых объектов (т.е имеет место случай “телесной трассировки”);
2. Проектируемая трасса задается своими начальными и конечными точками в системе координат, а также заданной площадью канала (или графиком изменения площадей по длине канала);
3. Проектирование канала производится с учетом уже существующей компоновки оборудования, которую мы менять не вправе;

4. Считаем, что наилучшей из возможных трасс будет та, которая соединяет заданные точки входа и выхода линией минимальной длины;

5. При проектировании трассы не производится гидравлический расчет течения жидкости или газа, считается, что удовлетворение заданным требованиям по минимальному радиусу скругления канала R_{min} обеспечивает по нему стабильное протекание жидкости или газа. Требование обеспечения R_{min} является приоритетным даже по отношению к требованию минимальной длины трассы – возможно, что для обеспечения R_{min} трассу придется искусственно удлинить;

6. При проектировании канала не учитывается взаимного влияния соединительных трасс (например теплового или электромагнитного). Считается, что исключение влияния соединительных трасс друг на друга или на уже размещенное оборудование исключается как экранировкой соединительных трасс, так и заданием минимального расстояния между проектируемыми трассами и уже размещенным оборудованием (δ – окрестности).

Необходимо отметить, что заданной физической постановке задачи соответствуют случаи проектирования трасс в авиационной технике, причем в качестве объектов проектирования могут выступать как гидро- пневмосистемы летательных аппаратов (рисунок 2.7), так и жгуты электропроводки (рисунок 2.8).



Рисунок 2.7 – Гидросистема самолета “Сухой Суперджет 100”

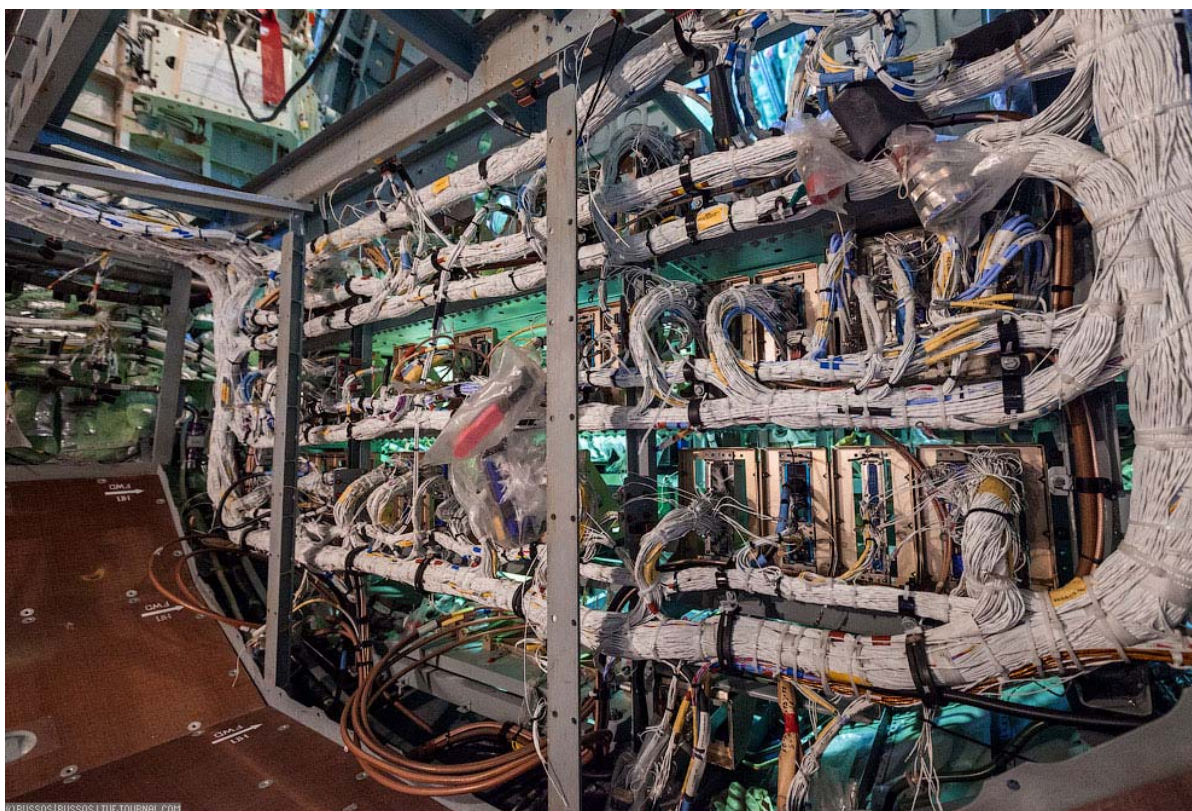


Рисунок 2.8 – Жгуты электросистемы самолета “Сухой Суперджет 100”

2.2 Математическая постановка задачи исследования

С математической точки зрения задача компоновки авиационной техники (впрочем, как и любой другой) может быть сформулирована как оптимизационная задача следующего вида.

Пусть имеется N компонуемых объектов T_i ($i=1, \dots, N$) и область размещения Ω . Требуется разместить эти объекты с учетом заданных ограничений в области Ω таким образом, чтобы функция цели компоновки $\Phi(X)$ достигала экстремума, т.е. определить

$$\text{Extr } \Phi(X) \text{ при } X \subset \Omega$$

где X - некоторая переменная, определяющая параметры размещения.

Таким образом, математическая постановка задачи размещения включает 3 компонента:

- 1) Выбор функции цели $\Phi (X)$.
- 2) Выбор переменной X .
- 3) Выбор и формализация ограничений.

Основным геометрическим условием рационального размещения объектов является условие рационального размещения в области Ω некоторого множества компоуемых объектов $\{T_i\}_1^n$ (где n достаточно велико) или наибольшего количества множеств объектов $\{\{T_i\}_1^n\}_1^m$ (где m достаточно велико). В данном случае приходится говорить о рациональном, а не об оптимальном размещении объектов, т.к. в силу астрономического количества вариантов компоновки (ее многовариантности) достижение глобального экстремума функции цели компоновки практически исключается [146].

Очевидно, что в геометрическом плане основным критерием оптимизации размещения является оптимизация коэффициента заполнения пространства K_V . Коэффициент K_V (иногда его называют коэффициентом плотности компоновки) представляет собой отношение

$$K_V = \frac{\sum_{i=1}^n V_{к.о.}}{V_{отс.}}$$

где $\sum_{i=1}^n V_{к.о.}$ - сумма объемов n скомпонованных объектов,

$V_{отс.}$ - объем отсека, в котором производится компоновка.

Условие максимальной плотности компоновки записывается в виде выражения

$$\begin{aligned} & Extr \Phi (X) \quad \text{при } X \subset \Omega & (2.1) \\ & \begin{matrix} K_V \rightarrow 1 \\ L_X \rightarrow \min \end{matrix} \end{aligned}$$

Выражение (2.1) представляет собой пожелание приблизить объем скомпонованных объектов к общему объему отсека, тем не менее, не является самым удобным для проведения дальнейших компоновочных расчетов. В нашем конкретном случае (телесной трассировки) смысл выражения (2.1) означает пожела-

ние провести соединительную трассу таким образом, чтобы она не только позволила выполнять свои основные функции (передачу необходимого потока газа, жидкости, тепла и т.п.) между двумя заданными точками пространства, но и обладала следующими дополнительными свойствами:

- обеспечивала бы условия взаимного непересечения с уже скомпонованными объектами и другими областями запрета;

- имела бы минимально возможную в данных условиях суммарную протяженность трассы L_{Σ} (что снизит ее гидравлическое сопротивление и массу);

- обеспечивала бы заданную конструктором плавность тока, что задается дополнительным техническим требованием – минимальной кривизной главной направляющей линии соединительной трассы и заданный график площадей.

Поэтому дальнейшей детализацией выражения (2.1), необходимой для оптимизации по K_V , является переход от минимизации по объему к минимизации по расстоянию L_{Σ} между объектами с обязательным соблюдением вышеописанных дополнительных ограничений. Оптимизация по K_V достигается максимально компактным (в идеале - плотным) размещением компокуемых объектов, фундаментальное понятие которого введено Ю.Г. Стояном и Н.И.Гилем [184, 185].

2.3 Анализ исследований в области автоматизации проектирования трасс

Хотя даже такие задачи трассировки являются достаточно трудноформализуемыми и труднорешаемыми из-за своей многоэкстремальной природы, они решены в работах проф. Ю.Г.Стояна и его учеников (Смеякова С.В., Аристовой И.В.) [184, 187, 174]. Однако в этих работах задача трассировки может быть сведена к задаче поиска минимального пути трассы по манхеттенской метрике. Это противоречит одному из главных наших условий физической постановки задачи

трассировки – линия трассы должна иметь минимальное количество плавных изгибов радиуса, не менее заданного. Автоматизация проектирования связей между механизмами, которые в определенном смысле также можно рассматривать как трассы, описана в монографии Арайса Е. А., Дмитриева В. М. [7].

Однако вопросы проектирования трасс именно по манхеттенской метрике, по нашему убеждению, являются наиболее разработанными в теории геометрического моделирования трассировок. Это связано с чрезвычайной практической важностью решения этих задач при автоматизированной разводке печатных плат и больших интегральных схем. Основополагающими в этой области являются теоретические исследования Абрайтиса Л.Б. [1], Базилевича Р.П. [12, 13], Лузина С.Ю., Петросяна Г.С. и Полубасова О.Б. [91, 92, 90, 93, 87, 142], Петренко А.П. и Тетельбаума А.Я. [138], Селютина В.А. [163], Стешенко В.Б. [181] и др. а также зарубежных ученых Эдсгера Вйбе Дэйкстры (E. W. Dijkstra) [218], Джуда Перл (Judea Pearl) [238,239,240], Ира Поль (Ira Pohl) [242,243], Даниэля Деллинга (Daniel Delling)[217], Питера Харта (Peter E. Hart) [224] и др.

Все эти исследования позволили разработать промышленные системы автоматизированной трассировки – P-CAD, ТороR и другие [68, 71, 151]. Типичный пример результата разводки в таких программах приведен на рисунке 2.9. Из этого рисунка видно, что трассировка в таких системах осуществляется либо по манхеттенской метрике, либо под углами 45° , либо под прямыми углами при переходе проводника на следующий слой (современные электронные платы как правило многослойные). Впрочем, как уже отмечалось во введении, такие резкие повороты трассы на движение электронов по ним никак не влияют.

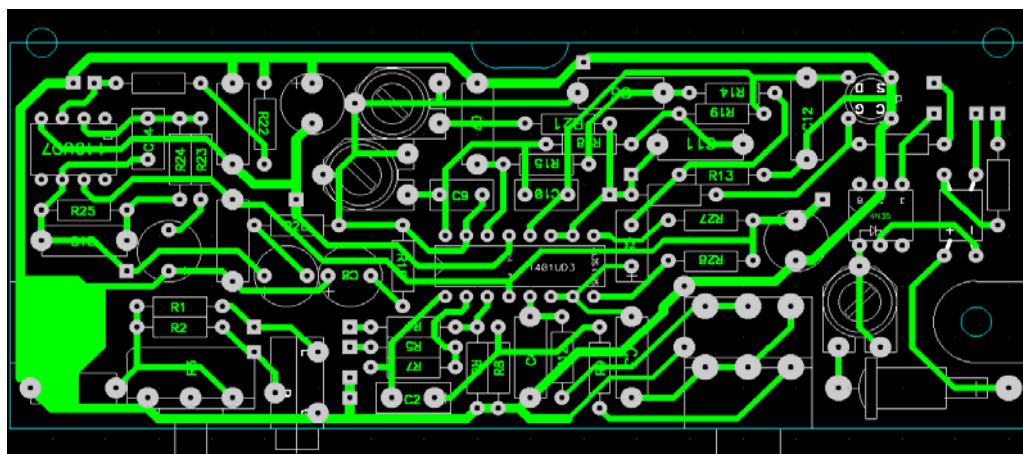


Рисунок 2.9 – Пример трассировки многослойной печатной платы в системе P-CAD

Таким образом, имеющиеся научные наработки в этой области напрямую не решают поставленную задачу – проектирование плавных трасс между уже скомпонованными объектами. Отметим и другое ограничение известных моделей трассировки – собственный размер трассы считается пренебрежимо малым по сравнению с размерами объектов, которые они соединяют. В ряде случаев такое допущение приемлемо с технической точки зрения, но в наших задачах проведения воздушных и гидротрасс между плотно компонованными объектами требует значительного усложнения геометрической модели трассировки. В технике трассировки, в которых размеры трасс соизмеримы с размерами скомпонованных объектов, называются *телесными*. Поэтому в дальнейшем с физической точки зрения будем рассматривать наш случай, как случай *телесной трассировки*. На практике, это проектирование трубопроводов, воздухопроводов и других элементов транспортных систем и, прежде всего, авиационной техники.

С точки зрения геометрического моделирования телесная трассировка считается особым и значительно более сложным видом трассировки. Она исследована в трудах Стояна Ю.Г., Аристовой И.В., Смелякова С.В. [8, 174, 9] и Калинина Б.В. [68, 69], а также зарубежными учеными Albano A.[211], Kubiak P. [244] и др.

2.4 Обоснование выбора метода геометрического моделирования телесной трассировки

Из проведенного выше анализа методов геометрического моделирования соединительных трасс следует, что, с одной стороны, известны методы и геометрические модели точного описания каналовых поверхностей (даже с учетом их дифференциально-геометрических характеристик), что, однако, практически исключает их автоматизированную компоновку ввиду исключительно сложного аналитического описания. С другой стороны, известны методы и геометрические модели автоматизированной компоновки объектов, которые применимы лишь к объектам простых геометрических форм (примитивам и композициям примитивов). Таким образом, для проектирования каналовых поверхностей между двумя объектами в транспортной технике необходимо смоделировать гладкий путь, учитывающий не только уже ранее скомпонованные объекты, но и обеспечивающий заданную плавность тока. Сложность решения этой задачи заключается в том, что при достаточно сложных геометрических формах размещенных объектов геометрическая форма коммуникаций окажется еще более сложной.

Приведенный выше анализ научных публикаций в этой области показал, что решение задачи телесной трассировки в вышеописанной постановке (т.е. с учетом требований компоновки и плавности одновременной) отсутствует. Принципиальным отличием в нашем подходе и подходе других исследователей является то, что если раньше канал проектировали по заданным инженерно-геометрическим характеристикам, а потом его уже размещали, то у нас наоборот – мы пытаемся спроектировать канал с заданными характеристиками, «вписанный» в уже существующую компоновку. Очевидно, что в нашей постановке задача не всегда имеет допустимые решения.

Отдельной строкой можно рассматривать возможности, предоставляемые современными САД-системами, позволяющими своими инструментальными

средствами выявить случаи взаимного пересечения уже скомпонованных объектов в созданной виртуальной моделировке. Но в этом случае речь идет не об автоматизированной компоновке, а о проверке уже сгенерированной с учетом опыта и интуиции проектанта варианта компоновки.

Таким образом, при решении поставленной задачи нам приходится выбирать рациональную геометрическую модель из дилеммы – что лучше – точная геометрическая модель, автоматическая компоновка которой невозможна, или грубая геометрическая модель, допускающая возможность автоматизированной компоновки. Попытаемся при решении задачи автоматизированной компоновки каналовых поверхностей выбрать компромиссное решение среди моделей в классе методов аналитической аппроксимации (см. рисунок 1.9).

Для решения поставленной задачи нам кажется предпочтительным использование рецепторных моделей, дискретизирующих пространство. В основу рецепторного метода (известного также как «матричный», «бинарный», «перечисления элементов пространства» и т.д.) положено приближенное представление геометрического объекта в поле или пространстве рецепторов. Для плоского случая поле рецепторов представляет собой однородную прямоугольную сеть $m \times n$, каждая клетка которой рассматривается как отдельный рецептор, который может иметь два состояния – «0» или «1». Математически рецепторная геометрическая модель описывается множеством $A = \{a_{ij}\}$, где

$$a_{i,j} = \begin{cases} 1, & \text{если рецептор возбужден,} \\ 0, & \text{если рецептор не возбужден} \end{cases}$$

Рецептор считается невозбужденным, если через него не проходит граница объекта и он не принадлежит внутренней области (рисунок 2.10).

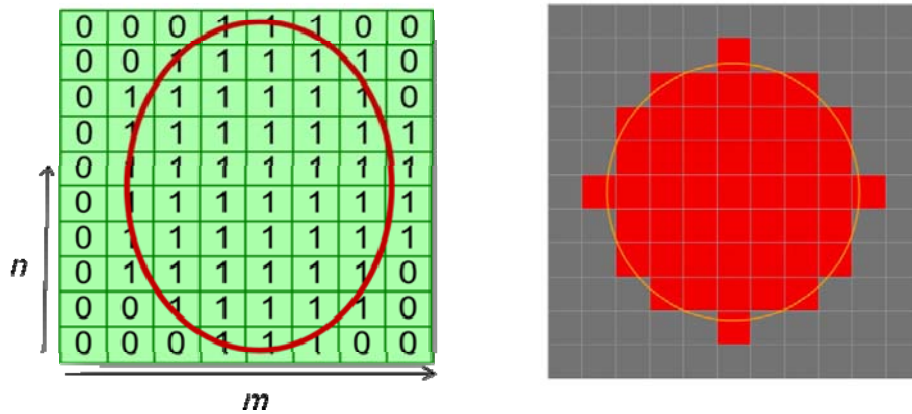


Рисунок 2.10 - Рецепторная модель 2D-тела

Трехмерные объекты описываются трехмерной матрицей $A = \{a_{i,j,k}\}$ размерностью $m \times n \times p$ (рисунок 2.11).

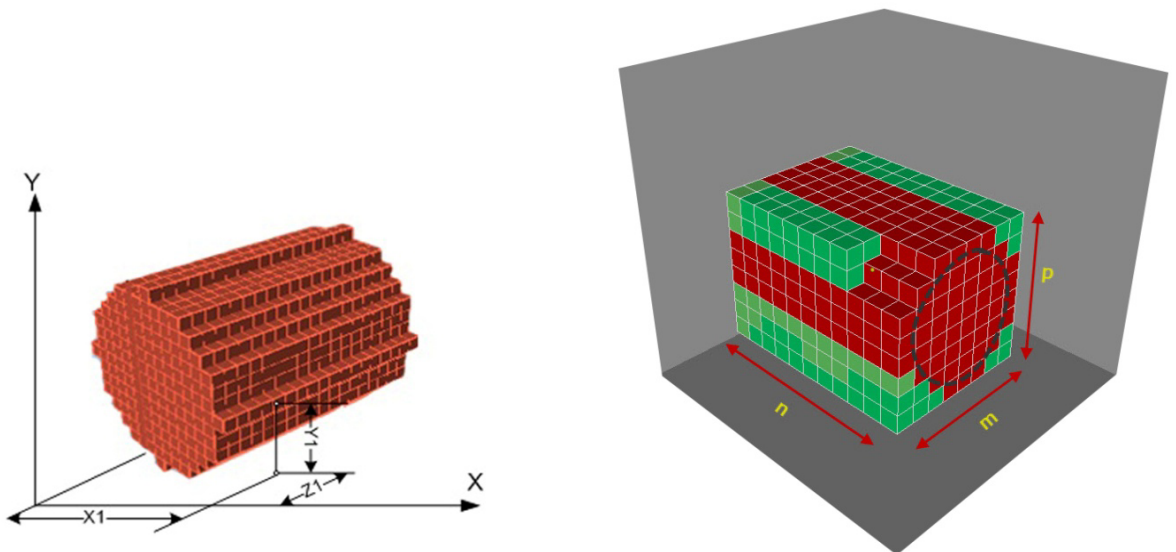


Рисунок 2.11 - Рецепторная модель 3D-тела

Существенной проблемой при решении задачи трассировки является обход препятствий, в качестве которых выступают уже скомпонованные объекты или коммуникации между ними. Большим преимуществом рецепторного подхода является легкость обнаружения препятствия по коду рецептора (0 или 1). Наипро-

стейшим подходом к проблеме является игнорирование препятствий до столкновения с ними. Такой алгоритм будет выглядеть примерно так: выбрать направление для движения к цели и, пока цель не достигнута и направление свободно для движения (рисунок 2.12).

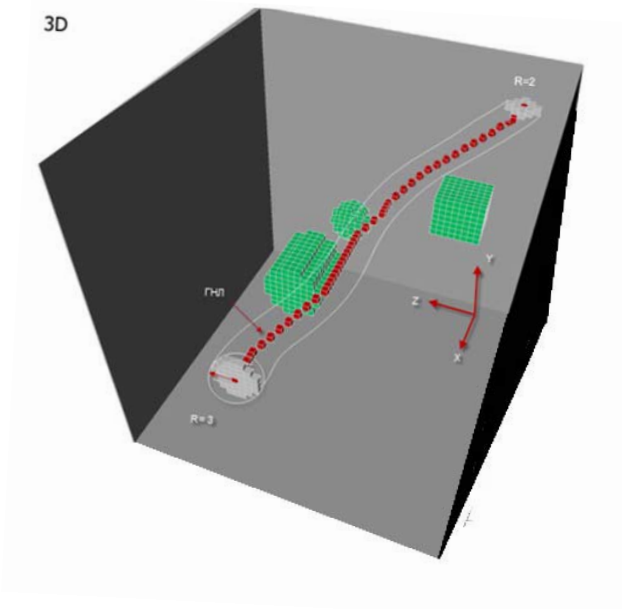


Рисунок 2.12 – Принцип обхода препятствий при построении каналовой поверхности рецепторным методом

Очевидно, что точность описания геометрической формы объекта зависит от выбранной нами дискретности рецепторной матрицы. Такой метод геометрического моделирования был предложен в начале 70-х годов прошлого века белорусским ученым Зозулевичем Д.М. [57, 58, 59, 60], но в те он годы не получил распространения из ограниченных возможностей ЭВМ по памяти и быстродействию. Хотя им и коллективом его сотрудников и были решены этим методом отдельные прикладные задачи, на ЭВМ тех лет с 16-битной архитектурой и объемом оперативной памяти 32...128 килобайт было невозможно рассчитывать на эффективное использование рецепторных моделей.

В дальнейшем, в связи с развитием производительности вычислительной

техники, рецепторные геометрические модели нашли свое практическое применение. Исследование и разработка рецепторных геометрических моделей для различных случаев применения была проведена в работах отечественных ученых Горелика А.Г. [31, 32, 33], Герасименко Е.П. [27, 28, 29], Клишина В.В. [70], Корн Г.В. [74, 75, 76], Рогозы Ю.А. [153, 154], Пащенко О.Б. [134, 135, 136], Толока А.В. [190, 191, 36], Ситу Лина [169, 170, 171], а также ряда иностранных авторов – Гаргантини И. (Gargantini I.) [220, 221, 212], Реквишы А.А.Г., (Requcha A.A.G.) [245, 246, 152] и ряда других [220, 221, 237, 249].

Здесь же следует отметить очень близкие по идеологии исследования Наджарова К.М. [109, 110, 108], Роткова С.И. [155, 156, 157] и др., в которых в качестве элементарного объекта формы выступает не классический рецептор в виде куба или параллелепипеда, а более сложные фигуры – например гексоэксаэдр.

Подводя итог вышесказанному, нами предлагается использовать для решения поставленной задачи рецепторные геометрические модели, т.е. разбиение компоновочного пространства на отдельные области в виде параллелепипедов (рецепторы - по западной терминологии **Voxel**), для каждой из которых в памяти компьютера присваивается значение «0» если она является свободной от размещенных объектов и доступна для размещения и «1» - если область уже занята размещенным объектом или коммуникацией к нему. Этот метод, сравнивая значения рецепторов, позволяет легко определять пересечение объектов.

По своей геометрической сущности рецепторный метод, который мы предполагаем использовать для решения поставленной задачи, является частным случаем метода аналитической аппроксимации объектов, который используются для описания трехмерных объектов, включающих сложные поверхности второго и более высоких порядков. Поскольку вычислительная обработка таких поверхностей затруднена, они аппроксимируются участками поверхностей более низкого порядка (плоскостями, цилиндрами и т.п.).

Рецепторный метод имеет как свои достоинства, так и недостатки. К очевидным недостаткам следует отнести дискретность модели и потребность в больших объемах памяти СВТ для своей реализации, но сейчас увеличение памяти СВТ до любых объемов не представляет сложности ни технической, ни экономической. Другой сложностью является то, рецепторная геометрическая модель (РГМ) никогда не является изначальной. Размещаемые и уже размещенные изделия описываются конструктором, как правило, параметрическими геометрическими моделями (т.е. задающими вид объекта и его параметры - сфера радиусом R , параллелепипед размерами $a \times b \times c$ и т.д.). Поэтому возникает необходимость в дополнительном программном модуле «Параметрическая модель» ↔ «Рецепторная модель». Но вопросы такого преобразования решены в работах Г.В.Корн [76] и Ситу Лина [171].

Безусловным же достоинством РГМ является уникальная легкость определения условия взаимного непересечения (УВН) уже размещенных объектов – если значение конкретного рецептора «0», то он доступен для размещения, если «1» - то он уже кем-то «занят». Именно эти причины побудили в последние годы интерес к РГМ как в странах СНГ, так и на Западе (там они называются «воксельными» - элемент объема – по аналогии с «пиксельными» - элементами плоскости).

2.5 Выбор направления разработки алгоритмов телесной трассировки

Особенностью научного подхода к разработке методов и алгоритмов проектирования трасс с использованием рецепторных моделей в данной диссертации является то, проектирование трассы рассматривается как задача искусственного интеллекта (ИИ). В соответствии с таким подходом, ГНЛ трассы представляется как путь, который от начальной до конечной точки предстоит выбрать и пройти

“одинокому путнику”, в качестве которого выступает одиночный рецептор. Уже размещенные объекты воспринимаются им как внешние препятствия, которые необходимо обойти с минимальными для себя потерями (т.е. с минимальным дополнительным для себя путем для обхода). При этом “одинокому путнику” (т.е. рецептору) при движении запрещено резко менять траекторию движения – таким образом реализуется заданная плавность траектории. На наш взгляд, такую ситуацию достаточно хорошо иллюстрирует ранее приведенный рисунок 2.12.

Так как перед нами стоит задача проектирования телесной трассировки, то для ее проектирования мы одним рецептором не обойдемся. Если требуется спроектировать трассу постоянного сечения, то будем считать, что у нас “путешествует” группа неразделимых между собой рецепторов, образующих при движении трассу заданного размера как совокупность мгновенных положений рецепторов по всей длине пути. Правда, это резко сужает возможности выбора пути по сравнению с одиноким рецептором – многие ранее доступные пути становятся недоступными. Говоря образным языком, там, где пролезала мышь не всегда пролезет корова, а там, где может пролезть корова, может не пролезть слон. Таким образом, не исключено, что допустимый вариант трассировки в данной компоновочной ситуации вообще отсутствует.

По нашему мнению, наиболее близко используют подобный подход следующие известные алгоритмы трассировки [167,173]:

- Алгоритм Дейкстры
- Алгоритм A^* «А звездочка»

Принцип работы этих алгоритмов близок к методологии рецепторных геометрических моделей, большим преимуществом которых является легкость обнаружения препятствия по коду рецептора (0 или 1). Наипростейшим подходом, реализованном в этих алгоритмах, является игнорирование препятствий до столкновения с ними. Такой алгоритм будет выглядеть примерно так: выбрать направ-

ление для движения к цели и, пока цель не достигнута и направление свободно для движения. Структура выбора направления движения определяется правилом:

двигаться туда

иначе

выбрать другое направление в соответствии со стратегией обхода.

Это позволяет утверждать, что в этих алгоритмах заложены элементы искусственного интеллекта (ИИ), так как решение выбирается по предикативному принципу “Если” – “То”.

В работах Дейкстры (Dijkstra, E. W.)[218], Дональд Эрвин Кнута (Donald Ervin Knuth)[227], Томас Кормена (Thomas H. Cormen)[216], Чарльза Лейзерсона (Charles Leiserson)[216] и др. [220, 225, 230] проанализированы различные стратегии обхода препятствий (эвристики), основанные как на случайном поиске, так и на алгоритмах искусственного интеллекта. Каждый из них имеет как свои ограничения, так и области предпочтительного применения. Примеры работы различных алгоритмов трассировки приведены на рисунке 2.13.

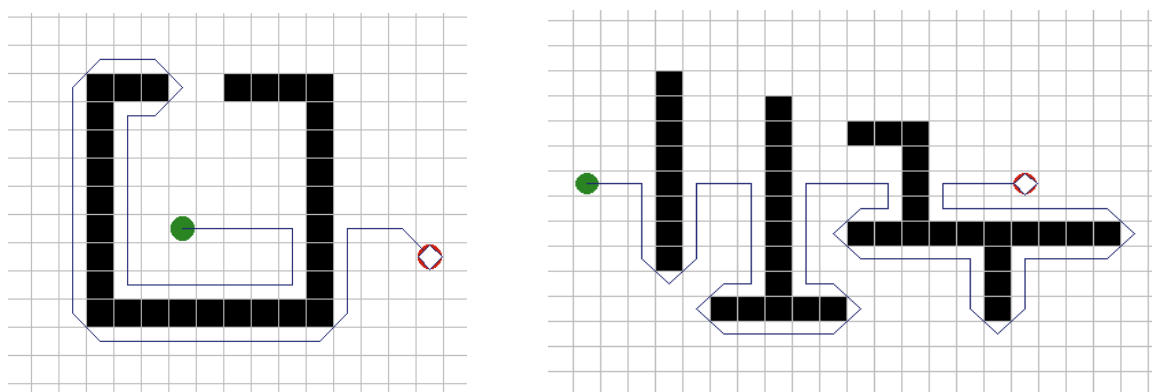


Рисунок 2.13 - Примеры трассировки при обходе различных препятствий (в 2D постановке) по алгоритму Дейкстры

Из рисунка 2.13 видно, что хотя известный алгоритм Дейкстры плоской трассировки смогли пройти путь от начальной до конечной точек, они сделали это не рационально. Привязанные к препятствиям (областям запрета), в соответствии

с заложенными в них эвристиками они держались их “как слепой стены”, сделав при этом чрезвычайно много лишних движений и многократно неоправданно изменяя направление перемещения.

Рецепторный алгоритм A^* действует более разумным способом и обычно используется для нахождения оптимального кратчайшего пути. Этот алгоритм гарантирует целостность и оптимальность, но такой путь не способен учитывать заданный зазор (δ –окрестность) - трасса может проходить слишком близко к областям запрета. Таким образом, поиск пути трассы по алгоритму A^* дает наилучшие результаты, но не обеспечивает решение поставленной задачи.

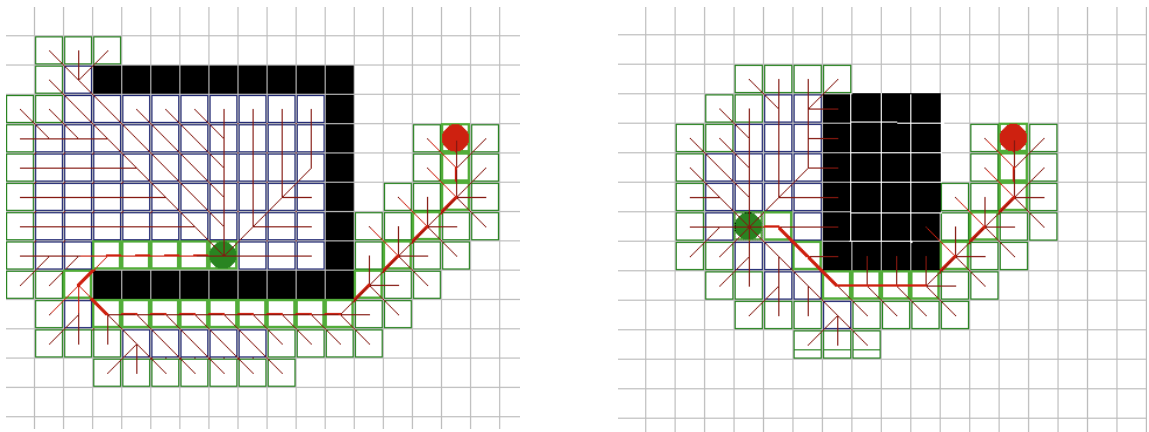


Рисунок 2.14 - Примеры трассировки при обходе различных препятствий (в 2D постановке) по алгоритму A^* .

Проведенное исследование показало, что из известных методов наиболее эффективным решением является использование известного алгоритма A^* , который, однако, в существующем виде не может быть использован для решения поставленной задачи – телесной трассировки. Причиной этого являются следующие некоторые его функциональные ограничения в существующем виде:

1. Проложенная с его помощью трасса, как правило, содержит резкие перемены направления движения и в принципе не удовлетворяет никаким требованиям плавности.

2. Нет способов, позволяющих строить прохождение трассы на заданном расстоянии от областей запрета (т.е. соблюдения δ –окрестности).

3. В настоящем виде алгоритм A^* не способен учитывать заданные изменения площади или объема трассы по мере движения по ней.

Все это говорит о том, что известный алгоритм A^* нуждается в существенной модификации, но может быть взят за основу как прототип для разработки новых алгоритмов, позволяющих решить поставленную в данной диссертации задачу – разработку моделей и алгоритмов пространственной телесной трассировки.

ЗАКЛЮЧЕНИЕ И ВЫВОДЫ К ГЛАВЕ 2

Рассмотрение научной и технической литературы по вопросам автоматизации компоновки показало, что эта тема является актуальной и исследования в этой области ведутся достаточно давно. Однако в настоящее время решены лишь такие частные вопросы автоматизированной компоновки, как компоновка трасс в многослойных печатных платах радиоэлектронной аппаратуры или интегральных микросхемах, автоматическое размещение (регулярное или нерегулярное) плоских (2D – объектов) сложной формы или относительно простых пространственных (3D – объектов) относительно простой формы (примитивы или композиции примитивов). В любом случае решение задач автоматизированной компоновки связано с решением переборных многоэкстремальных задач. Использование же инструментальных средств современных САД систем твердотельного моделирования не решает задач автоматизированной компоновки, а позволяет лишь проанализировать на отсутствие взаимного непересечения уже готовые компоновочные решения. Все это позволяет сформулировать следующие выводы:

1. Описана физическая постановка задачи исследования – проектирования телесной трассировки с учетом присущей ей конструктивно-технологических ограничений.

2. Сформулирована математическая постановка задачи исследования как оптимизационная многоэкстремальная задача математического программирования.

3. Проведенный анализ научной и технической литературы показал, что известна библиотека методов и алгоритмов, позволивших создать промышленные системы автоматической трассировки радиоэлектронной аппаратуры, однако непригодных для решения поставленной задачи телесной трассировки из-за невозможности выполнения физических ограничений на прохождение трассы.

4. Имеется библиотека методов и геометрических моделей, позволяющих автоматизировать процесс компоновки, однако допускающих автоматическое размещение лишь компонуемых объектов простейших форм (примитивов и композиций примитивов) и непригодных для автоматизированной компоновки столь сложных геометрических объектов, как каналовые поверхности.

5. Компромиссным вариантом между точностью геометрической модели и возможностью ее автоматизированной компоновки является использование рецепторных геометрических моделей, в которых уникально просто решается вопрос выявления случаев взаимного пересечения компонуемых объектов.

6. Представляется перспективным направлением исследований в области автоматизированной компоновки каналовых поверхностей разработка геометрических моделей и реализующих их алгоритмов на основе интеллектуальных рецепторных алгоритмов трассировки Дейкстры и алгоритма A^* , которые однако в существующем виде нуждаются в существенной модификации.

3 РАЗРАБОТКА И ИССЛЕДОВАНИЯ РЕЦЕПТОРНЫХ ГЕОМЕТРИЧЕСКИХ МОДЕЛЕЙ ФОРМИРОВАНИЯ КАНАЛОВЫХ ПОВЕРХНОСТЕЙ

3.1 Исследование и анализ известных алгоритмов трассировки, основанных на рецепторных геометрических моделях

Как уже отмечалось в предыдущих разделах, в научной и технической литературе описано достаточно большое количество методов и алгоритмов нахождения кратчайшего пути между двумя точками, использующих плоские рецепторные модели. Такие алгоритмы используются во многих технических приложениях и для различных видов деятельности – проектирование транспортных систем, сетей коммуникаций, в компьютерных играх, а также при проектировании и моделировании различных технических систем.

Общей чертой всех этих алгоритмов является то, что их конечной целью является построение трассы из начальной (стартовой) позиции в конечную. Каждый алгоритм в своей работе использует стек и очередь для хранения данных о каждом узле и его потомках, что сильно увеличивает скорость и эффективность программы. Обычно все алгоритмы нахождения пути пытаются найти лучший и наиболее эффективный путь к цели. Некоторые алгоритмы основаны на приемах эвристики, моделирующих творческое, неосознанное мышление человека. Применение эвристических приемов позволяет значительно сократить перебор вариантов и, тем самым, ускорить решение задачи.

Использование эвристических алгоритмов превращает задачу трассировки (расчета кратчайшего пути) в задачу искусственного интеллекта, при которой развивается поиск взаимосвязи между причиной и следствием, появляется возможность самообучения алгоритма. Количественной мерой эффективности выбора при таком подходе являются “стоимость пути” и количество пройденных узлов.

Процесс оптимизации траектории заключается в процессе понижении стоимости пути, уменьшения числа пройденных узлов и сортировки узлов внутри стека или очереди.

Существует множество различных подходов к решению задач и нахождения пути, но, в силу специфики нашей задачи, мы не будем подробно описывать каждый из них. Задачу нахождения пути можно разделить на два основных подхода: **ненаправленная и направленная трассировки** (рисунок 3.1).



Рисунок 3.1 - Две основных подхода к нахождению пути

Ненаправленный подход можно сравнить с мышью, слепо бегающей по лабиринту в поисках выхода. Она не тратит время на планирование пути, а тратит всю энергию на обход территории. При этом мышь может и не найти выход, затрачивая слишком много времени на обследование тупиков. Существуют две основных модификации ненаправленного подхода к решению поставленной задачи, которые повышают эффективность поиска. Это известные алгоритмы *поиска в ширину* и *поиска в глубину*, описанные в [1].

Поиск в ширину рассматривает предполагаемую область, как узлы большого связанного графа. При этом поиске исследуются все узлы, связанные с теку-

щим узлом, а затем, в свою очередь, проходятся узлы, связанные с пройденными. Таким образом, если возможный путь существует, то его можно найти с помощью алгоритма *поиска в ширину*. Кроме того, при наличии нескольких путей, этот поиск покажет наименее глубокий.

Метод *поиска в глубину* представляет собой подход, противоположный выше описанному. Сначала обходятся дочерние узлы каждой вершины, и тем самым, создается линейный путь к цели. Поиск в глубину полезен для нахождения пути от одной вершины к другой, независимо от того, является ли граф связным. При этом алгоритме обсчитывается основное дерево связанных графов. Этот метод используется метод поиска с возвратом в уже пройденные точки. Если поиск пути заходит в тупик, начинается обход узлов предыдущего уровня. Этот метод имеет преимущество в задачах со множеством решений, так как можно найти решение, исследовав лишь небольшую область заданного пространства.

Иллюстрация этих обоих подходов приведена на простом примере, показанном на рисунке 3.2.



Рисунок 3.2 Иллюстрация прохождения точек маршрута в пространстве и соответствующее ей дерево поиска от начальной точки (S) к целевой точки (G) при поиске в ширину (а) и поиске в глубину (б)

На рисунке 3.3 показано, как каждый из алгоритмов будет обходить граф для нахождения пути. Из него видно, что алгоритму *поиска в ширину* потребовалось 4 итерации, в то время алгоритм *поиска в глубину* нашел путь всего за 2 итерации. Это можно объяснить тем, что данная задача имеет несколько решений, и,

как было сказано выше, *поиск в глубину* является предпочтительней. Основным недостатком обоих подходов является то, что стоимость пути не учитывается. Таким образом, оба алгоритма могут быть использованы и эффективны в том случае, если не введен вес графа.

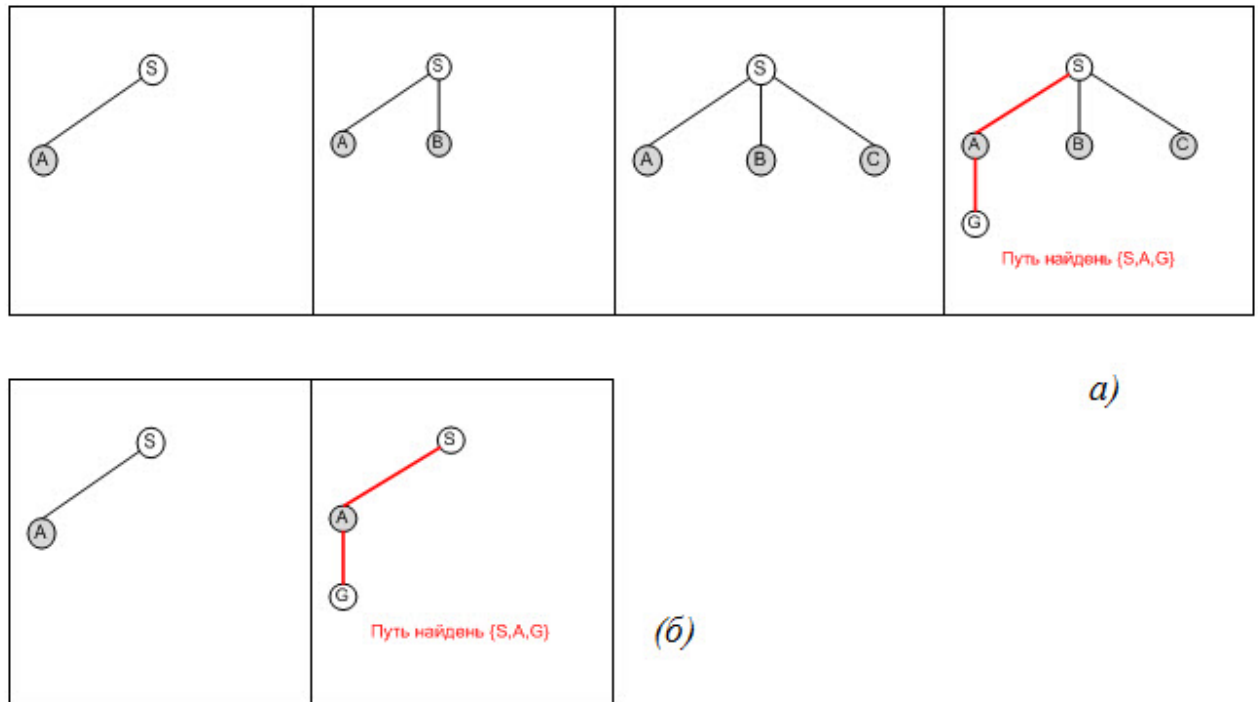


Рисунок 3.3 – Иллюстрация количества итераций при нахождении пути алгоритмом “поиск в ширину” (a) и алгоритмом “поиск в глубину” (б)

Значительно более перспективными для нашего исследования является использование алгоритмов, реализующих *направленный подход*. У направленных подходов к решению задачи поиска пути есть характерная особенность: они не слепо блуждают по лабиринту, а прежде чем выбрать следующую вершину траектории, производят оценка эффективности такого выбора для всех смежных узлов. Критерием эффективности оценки обычно выступает параметр “стоимости перехода” в смежный узел. Обычно в качестве параметра “стоимости пути” на рецепторной матрице используется расстоянием между узлами. Большинство используемых алгоритмов такого типа находят решение задачи, но не всегда самое эф-

фективное, то есть наикратчайший путь. Основными стратегиями нахождения пути с помощью направленных алгоритмов являются:

- Метод с *постоянной стоимостью* $g(n)$ ведет поиск, выбирая наименьшую стоимость пути до соседнего узла.
- Метод с *эвристическим поиском* $h(n)$ ведет поиск, оценивает стоимость от соседнего узла к цели. При этом стоимость пути изменяется по правилам, заложенным в эвристиках.

Одним из первых решений этой задачи стал "*алгоритм Дейкстры (Dijkstra's algorithm)*", предложенный голландским ученым Э. Дейкстрой в 1956 году и впервые опубликованный в 1959 году. Особенность этого алгоритма в том, что он находит кратчайший путь для всех вершин пространства поиска, а не для единственной целевой вершины используя *стратегию постоянной стоимости* для нахождения оптимального пути. В *алгоритме Дейкстры* обходятся все ближайшие "непомеченные" вершины из начальной точки, а не только целевая вершина [54, 163].

С математической точки зрения Алгоритм Дейкстры решает задачу о кратчайших путях из одной вершины для взвешенного ориентированного графа $G = (V, E)$ с исходной вершиной s , в котором веса всех рёбер неотрицательны ($\omega(u, v) \geq 0$ для всех $(u, v) \in E$).

В процессе работы алгоритма Дейкстры поддерживается множество $S \subseteq V$, состоящее из вершин v , для которых $\delta(s, v)$ уже найдено. Алгоритм выбирает вершину $u \in V/S$ с наименьшим $d[u]$, добавляет u к множеству S и производит релаксацию всех рёбер, выходящих из u , после чего цикл повторяется. Вершины, не лежащие в S , хранятся в очереди Q с приоритетами, определяемыми значениями функции d . Предполагается, что граф задан с помощью списков смежных вершин и стоимости пути между ними (рисунок 3.4).

Работа алгоритма Дейкстры осуществляется следующим образом. Инициализация работы алгоритма начинается с того, что метка исходной точки

А полагается равной 0 , метки остальных вершин — бесконечности. Это отражает то, что расстояния от a до других вершин пока неизвестны. Все вершины графа помечаются как непосещенные. Происходит перебор и анализ всех доступных вершин, а когда все вершины графа посещены, алгоритм завершается. В противном случае из еще не посещенных вершин выбирается вершина u , имеющая минимальную метку. Мы рассматриваем всевозможные маршруты, в которых u является предпоследним пунктом. Вершины, соединенные с вершиной u ребрами, назовем соседями этой вершины. Для каждого соседа рассмотрим новую длину пути, равную сумме текущей метки u и длины ребра, соединяющего u с этим соседом. Если полученная длина меньше метки соседа, заменим метку этой длиной. Рассмотрев всех соседей, пометим вершину u как посещенную и повторим шаг.

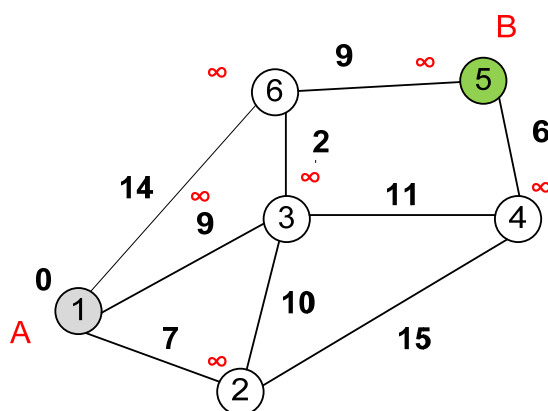


Рисунок 3.4 – Задание вершин и стоимости пути между ними в алгоритме Дейкстры

Другим алгоритмом является поиска пути является "**Лучший-первый (Best-First Search)**" [238, 239]. Его автор - Джуда Перл (Judea Pearl) описал направленный алгоритм поиска «Лучший — первый», основанный на эвристическом поиске. Данный алгоритм исследует граф пути анализируя наиболее перспективные узлы возможной траектории, выбираемые в соответствии с указанным правилом. В качестве правила выбора (эвристики) очередного узла прокладки траектории

выступает некоторая “эвристическая функция оценки” $f(n)$, которая, вообще говоря, может зависеть от описания трассы, описания цели и дополнительной информации о предметной области, описывающей особенности функционирования трассы [2, 3].

В основу его функционирования алгоритма Перла *“Лучший-первый”* положен принцип, противоположный алгоритму Дейкстры. Вместо того, чтобы из начальной точки пути обходить все ближайшие вершины, в алгоритме *“Лучший-первый”* выбирается вершина ближайшая к целевой. Но, так как мы не знаем точной траектории от текущей до целевой вершины, необходимо провести оценку возможной траектории. Поскольку такая оценка является эвристической, то алгоритм *“Лучший-первый”* не обеспечивает нахождение рационального, но не оптимального решения, что также неплохо.

Некоторые авторы модифицировали алгоритм поиска «Лучший — первый» специально для описания поиска с эвристикой, в которой пути, которые имеют лучшую эвристическую оценку, рассматриваются первыми. Этот специфический тип поиска называется *«Жадным поиском Лучший — первый»* [3].

В 1968 году для нахождения кратчайшего пути между двумя точками был предложен алгоритм поиска A^* *Peter Hart, Nils Nilsson and Bertram Raphael (Питер Харт, Нильс Нильссон и Бертрам Рафаэль)* [224], который широко используется и в наши дни, особенно в интерактивной индустрии развлечений. Он сочетает в себе подходы *алгоритма Дейкстры* и поиска *“Лучший-первый”*. Алгоритм A^* , также как и *алгоритм Дейкстры*, гарантирует нахождение оптимального решения (при условии отсутствия отрицательных весов и допущения эвристики), и, так как он стремится достичь целевой вершины, как в алгоритме *“Лучший-первый”*, но он не будет обходить столько же вершин, сколько рассматривается в *алгоритме Дейкстры*. Такой подход снижает требования к памяти и уменьшает время вычислений.

По принципу функционирования алгоритм нахождения пути A^* (произносится а-звездочка) – это *направленный* алгоритм, который не слепо ищет путь, подобно мыши в лабиринте, а оценивает наилучшее направление для движения, зачастую возвращаясь на предыдущий шаг для проверки альтернатив [183]. Алгоритм рассчитывает кратчайший (или лучший) путь из заданной точки в конечную точку с помощью сетки или узлов. Этот эвристический поиск сортирует все узлы по ранжированию наилучшего маршрута, проходящего через эти узлы. Таким образом, A^* не только находит путь между двумя вершинами (при условии, что он существует!), но стремится к относительно быстрому поиску кратчайшего существующего пути. Типичная формула эвристики выражается в виде:

$$f(n) = g(n) + h(n) \quad (3.1)$$

где: g - стоимость перехода из начального узла в текущий узел, то есть сумма всех величин, приписываемых пути от стартового до текущего узла;

h - обозначает эвристику, которая оценивает стоимость пути от текущего до целевого узла (обычно прямое расстояние от текущего узла до цели);

f - сумма g и h , обозначающая наилучшую оценку стоимости пути через текущий узел. Проще говоря, чем ниже значение f , тем эффективнее путь

До обеспечения работы алгоритма A^* необходима карта для поиска пути, представляющая собой разбиение области будущей трассы на отдельные точки или локации, называемые узлами. Это могут быть точки маршрута, многогранники сетки навигации или многогранники системы освещения области, которые используются для записи хода поиска. В данной работе будем использовать сетку рецепторной матрицы. Кроме координат расположения на карте узлы должны обладать тремя дополнительными атрибутами: *fitness (соответствие)*, *goal (цель)* и *heuristic (эвристика)*, обозначаемые, соответственно, f , g , и h . Путям между узлами могут приписываться величины, обычно обозначающие расстояния. Цель f , g , и h – количественно определить, насколько перспективным является путь до текущего узла. В работе алгоритма A^* функция эвристики оценки пути (обычно

обозначается $f(x)$), используются чтобы определить порядок прохождения узлов дерева. $f(x)$ – это сумма двух функций: функция оценки пути (обозначается $g(x)$), которая может быть эвристической (h), и допустимая эвристическая оценка расстояния до цели, $g(x)$ – это стоимость пути от начального узла до текущего.

Кроме этого, алгоритм A* создает два списка: *Открытый* и *Закрытый*. *Открытый список* хранит все узлы карты, которые еще не были исследованы, в то время, как в *закрытый список* заносятся уже пройденные узлы. Узел считается пройденным, когда алгоритм просмотрел все смежные с ним узлы. Поэтому узлы просто отмечают состояние и ход поиска.

Эвристика (Heuristic- H) - это функция, используемая в алгоритмах поиска для расчета оптимального пути при вычислениях. Иллюстрации этих параметров приведены на рисунке 3.5.

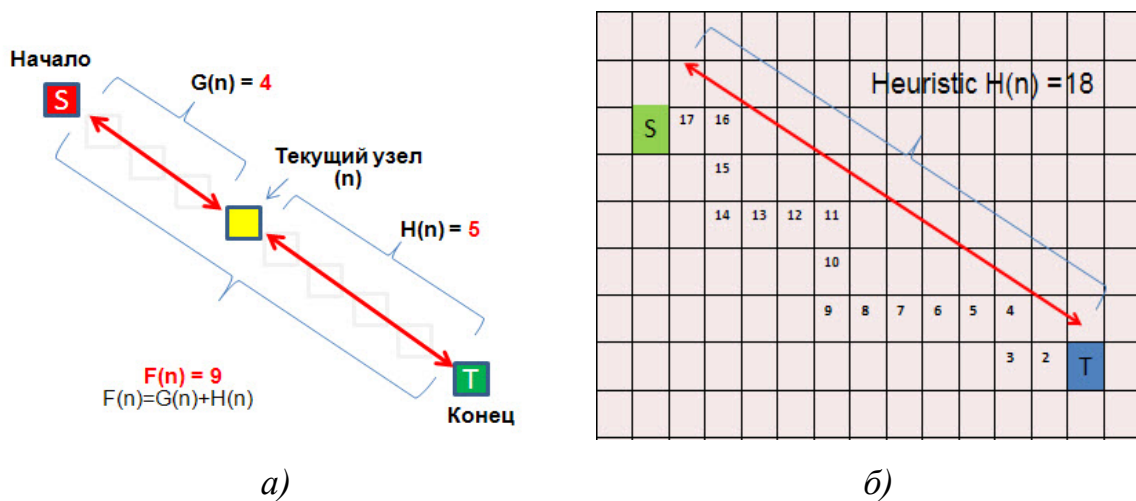


Рисунок 3.5 - Геометрическая иллюстрация параметров, определяющих работу алгоритма A*: а – функции оценки параметров пути $f(x)$ и $g(x)$; б – эвристика $H(n)$, задающая вычисленное направление движения

Поскольку алгоритм A* является наиболее совершенным на данный момент алгоритмом дискретной трассировки, приведем пример его работы на рисунке 3.6. Положим центральный узел (точка 2,2) начальной точкой (P), а смещенный серый узел (точка 0,1) – конечной точкой (E).

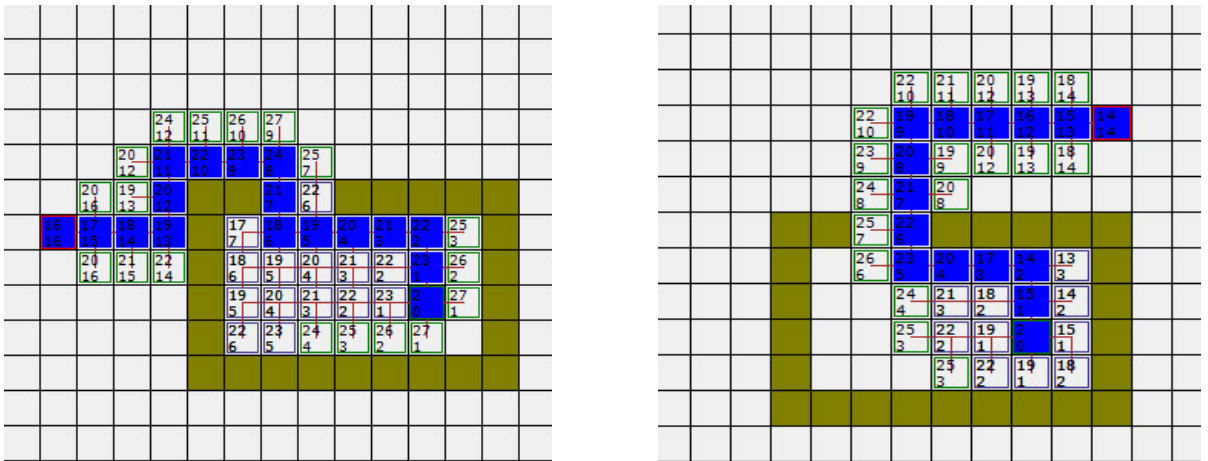


Рисунок 3.6 - Пример для иллюстрации работы алгоритма A*

Значение функции h рассчитывается каждый раз по-разному в зависимости от прикладной задачи. В данном примере величина h будет равна сумме вертикального и горизонтального расстояния до текущего узла (E). Таким образом, $h = |dx - cx| + |dy - cy|$, где (dx, dy) - узел назначения, а (cx, cy) – текущий узел.

Сначала занесем узел P(2,2) в открытый список, так как это единственный известный алгоритму узел. Результат показан в таблице 3.1.

Таблица 3.1 – Содержание открытого и закрытого списков на 1-ом этапе работы алгоритма A*

Открытый список	Закрытый список
P{2,2}	{Пусто}

У узла P(2,2) существует 8 соседних узлов, соответственно, (1,1), (2,1), (3,1), (1,2), (3,2), (1,3), (2,3), (3,3). Занесем каждый из этих узлов в открытый список, в случае если это не было сделано ранее. Затем проверим каждый узел из открытого списка, является ли он конечной точкой (1,0). Если нет, рассчитаем значение f по формуле $f = g + h$. Полученные результаты занесем в таблицу 3.2

Таблица 3.2 – Расчетные значения параметров на 2-ом этапе работы алгоритма А*

Узел	Значение g	Значение h	Значение f
(1,1)	0 (Узлы для прохождения)	1	1
(2,1)	0	2	2
(3,1)	0	3	3
(1,2)	0	2	2
(3,2)	0	4	4
(1,3)	0	3	3
(2,3)	0	4	4
(3,3)	0	5	5

Из таблицы 3. видно, что узел (1,1) обладает наименьшим значением f , а значит, это следующий узел, выбранный по алгоритму А*. Так как все смежные с P(2,2) узлы были просмотрены, занесем P(2,2) в закрытый список (как показано в таблице 3.3).

Таблица 3.3 – Содержание открытого и закрытого списков на 3-ом этапе работы алгоритма А*

Открытый список	Закрытый список
{ (1,1), (2,1), (3,1), (1,2), (3,2), (1,3), (2,3), (3,3) }	{ P(2,2) }

У узла (1,1) существует четыре смежных узла, соответственно, E(1,0), (2,1), (1,2), (2,2). Рассмотрим узел E(1,0), так как это единственный узел, не занесенный ни в один список. Все «соседи» (1,1) были просмотрены, поэтому занесем этот узел в закрытый список. Так как E(1,0) – конечный узел, путь к нему найден и занесен в закрытый список. Этот путь был найден методом обратного отслеживания узлов в закрытом списке от целевого узла к начальной точке {P(2,2), (1,1), E(1,0)}.

Поэтому алгоритм A^* всегда найдет кратчайший путь в случае, если такой путь существует [4].

Из всех этих рассуждений следует, что алгоритм поиска пути A^* является примером оптимального поиска по сравнению с алгоритмом «Лучший — первый». Два наиболее распространенных алгоритма направленного поиска пути используют одну или несколько этих стратегий. Если алгоритм Дейкстры используют *стратегию постоянной стоимости* для нахождения оптимального пути, то алгоритм A^* сочетает в себе обе стратегии, при этом минимизирую общую стоимость пути. **Таким образом, алгоритм A^* находит оптимальный путь и является более эффективным методом, чем алгоритм Дейкстры.** Алгоритм A^* служит основой для нахождения пути в компьютерных играх и является наиболее распространенным методом поиска пути в задачах трассировки для различных технических приложений.

Проведем сравнение эффективности наиболее совершенных известных алгоритмов трассировки, использующих направленный поиск [163, 54]. Иллюстрация такого сравнения приведена на рисунке 3.7. Для сравнения их эффективности мы используем ту же топологию графа, что была использована на рисунке 3.2 и соответствующее ей дерево с начальной точкой (S) и целевой вершиной (G). Однако, на этот раз обозначена стоимость пути вдоль определенного ребра графа.

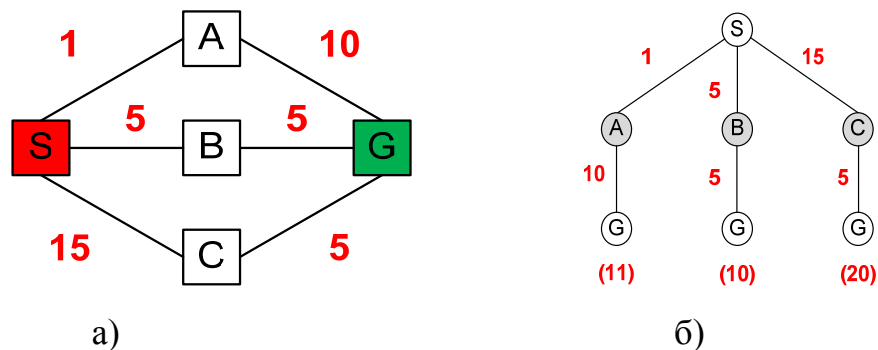


Рисунок 3.7 - Иллюстрация прохождения точек маршрута на рецепторной матрице и соответствующее ей дерево поиска от начальной точки (S) к целевой точки (G) для алгоритма Дейкстры (а) и алгоритма A^* (б)

Рисунок 3.8 иллюстрирует, как алгоритм Дейкстры и алгоритм A* будут осуществлять обход дерева для нахождения пути с учетом его стоимости на графе, изображенном на рисунке 3.6. В этом примере алгоритму Дейкстры потребовалось 3 итерации, в то время, как алгоритм A* нашел путь за две итерации, при этом найденный им путь является кратчайшим, то есть решение оптимально. На первом этапе поиска, показанном на рисунке 3.8, и алгоритму Дейкстры, и алгоритму A* требуется 3 итерации, так как каждый узел соединён с начальной вершиной (S). Еще одна итерация потребуется для следующего перехода. Таким образом количество итерация для алгоритма Дейкстры и A*, соответственно, шесть и пять. По сравнению с описанными методами *поиска в ширину* и *поиска в глубину* (см. рисунок 3.3), которым потребовалось соответственно пять и две итерации, направленным методам требуется большее число шагов, но оба подхода находят оптимальные пути в отличие от ненаправленных методов.

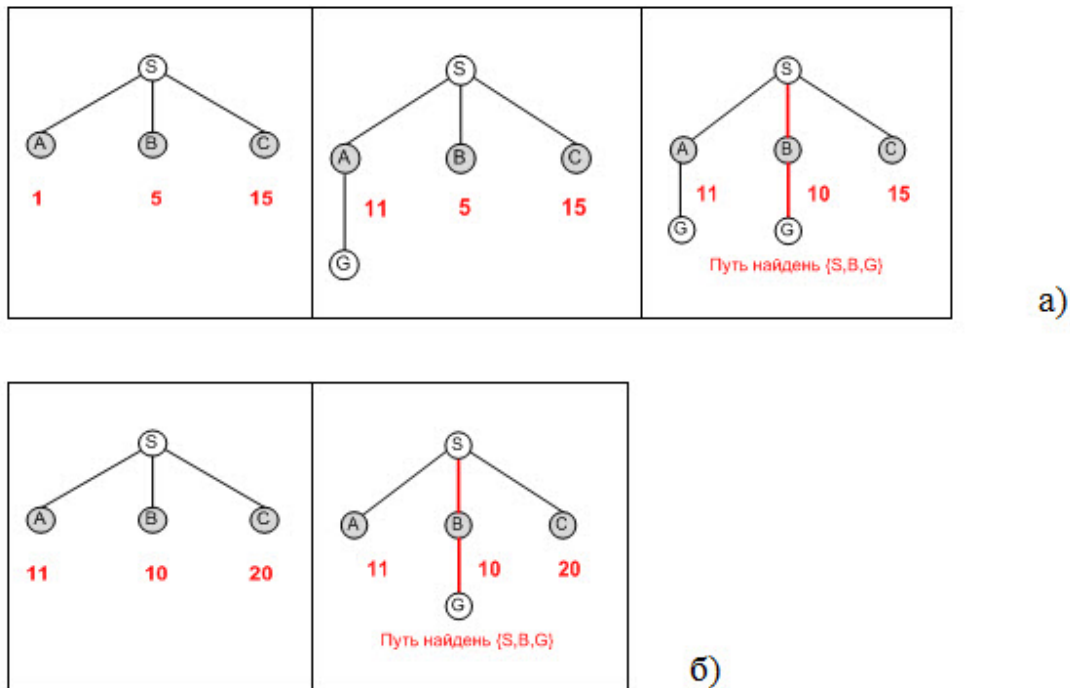


Рисунок 3.8 - Иллюстрация количества итераций при нахождении пути алгоритмом Дейкстры (а) и алгоритмом A* (б)

3.2 Выбор направлений модификации известных алгоритмов трассировки, основанных на рецепторных геометрических моделях

Приведенное выше исследование и анализ известных алгоритмов трассировки, основанных на рецепторных моделях, показал, что самым лучшим алгоритмом для поиска пути является алгоритм A^* . Его сильной стороной является гибкость, т.е. возможность решать достаточно широкий круг задач трассировки и получать при этом оптимальные решения. Суммарная оценка особенностей известных алгоритмов трассировки приведена в таблице 3.4.

Однако, при всех достоинствах алгоритма A^* , он в существующем виде не решает проблемы автоматизированного проектирования из-за присущих ему ограничений. Перечислим **ограничения алгоритма A^*** :

1. При большом количестве областей запрета или узлов, расположенных на рецепторной матрице большой размерности, требуется значительное количество вычислительных ресурсов (по мощности ЦПУ и памяти). Рецепторная матрица большой размерности неизбежно потребует тысячи и десятки тысяч записей в открытом и закрытом списках. На обработку этих списков может не хватить вычислительных мощностей компьютера.

2. Второе ограничение является следствием первого. В программах с последовательным выполнением операций обработка столь длинных списков может привести к значительному увеличению времени поиска решения. Время поиска еще более возрастет, если A^* ищет путь между уже размещенными объектами. Проведенные нами тесты показали, что высокая требовательность алгоритма A^* к вычислительным ресурсам иногда приводит к тому, что компьютер зависает до того, пока рациональный путь будет найден.

3. Для алгоритма A^* даже в случае отсутствия проблем с вычислительными ресурсами, его работа со списками может быть неэффективной. Это связано с тем, что эффективность работы алгоритма A^* сильно зависит от качества эвристиче-

ского приближения $h(n)$. Если величина h близка к истинному значению оставшегося пути, эффективность будет высокой. И, напротив, если значение h прогнозируется плохо, то эффективность работы этого алгоритма будет низкой.

Таблица 3.4 - Суммарная оценка особенностей известных алгоритмов трассировки

Алгоритмы	Возможно ли найти путь?	Требование к памяти	Скорость	Оптимальный путь	Область поиска
Поиск в ширину (Breadth-first Search)	да	высокое	медленно	нет	большая
Поиск в глубину (Depth-first Search)	да (нуждается в переделе глубины)	высокое	медленно	да	средняя
Алгоритм Дейкстры (Dijkstra's)	да	нормально	быстро (если область поиска маленькая)	да	средняя
Лучший-первый (Best-first Search)	да (надо $f(n)$)	ниже, чем Breadth-first Search	быстрее чем Breadth-first Search	да (если получена хорошая величина $h(n)$)	средняя
A звездочка (A*)	да	низкое	быстро	да	средняя

4. Из проведенного анализа следует, что наиболее совершенный рецепторный алгоритм A^* обычно используется лишь для нахождения оптимального кратчайшего пути. Этот алгоритм гарантирует целостность и оптимальность проложенной трассы, но ее траектория не способна учитывать заданное минимальное расстояние до препятствия - она может проходить слишком близко к областям запрета. Для решения этой проблемы были предложены полиномиальные методы приближения [202], что во много раз увеличивает сложность вычислений, а иногда и делает их невозможными.

5. И, наконец, алгоритм A^* не способен проектировать траектории с заданной степенью гладкости, так как такой задачи перед ним никогда не ставилось.

Именно последние два ограничения, столь необходимые для проектирования трасс в 3D – пространстве рецепторной матрицы, делают даже самый лучший алгоритм A^* непригодным для решения поставленной нами задачи. Разработка моделей и алгоритмов, позволяющих преодолеть указанные выше недостатки (особенно 4 и 5), и является главной темой настоящего исследования.

Справедливости ради необходимо отметить, что в научно-технической литературе известны попытки такой модификации алгоритма A^* , которые бы позволили с его помощью получать гладкие траектории [177, 181]. В научной литературе приведены данные, что сравнение результатов работы улучшенного таким образом алгоритма A^* по сравнению с классическим алгоритмом A^* показало, что сглаженные траектории оказываются значительно более длинными, но самое неприятное то, что при этом возможна блокировка (клинч) траекторий [175].

Для нахождения безопасного и быстрого пути был предложен метод Масатомо Канехары (Masatomo Kanehara) [226], использующий клотоиды для сглаживания с помощью R_{min} , который оптимизирует путь и учитывает допустимый уровень безопасности. Однако этот метод был создан для движения робота в 2D пространстве.

В работе Масатоми Канехары [226] было рассмотрено нахождение пути, основанное на рецепторной модели, в двумерном пространстве с использованием и улучшенного алгоритма A^* .

Алгоритм A^* был выбран в силу того, что он обеспечивает оптимальный и эвристический поиск. И, зачастую, он работает лучше остальных стратегий поиска.

Исследование алгоритма трассировки A^* , как наиболее совершенного для решения поставленной задачи, показывает, что решения поставленной диссертационной задачи необходима его модификация, позволяющая:

1. Получать в результате его работы гладкие траектории с конкретными дифференциально-геометрическими характеристиками (прежде всего с обеспечением заданного минимального радиуса кривизны R_{min});

2. Обеспечивать в случае необходимости прохождение трассы на заданном расстоянии от областей запрета (обеспечение δ - окрестности);

3. Нахождение пути, основанное на рецепторной модели, в двумерном и трехмерном пространстве с использованием и улучшенного алгоритма A^* и каналовых поверхностей. Были разработаны различные эвристики и вариации алгоритма A^* для решения проблемы поиска кратчайшего пути, удовлетворяющего ограничениям, определенным пользователем. Учитывать возможность изменения сечения трассы по длине траектории по заданному закону.

Целью использования каналовой поверхности является создание поверхностного моделирования, которое является неотъемлемой частью геометрического моделирования в 3D пространстве.

3.3 Разработка геометрической модели построения главной направляющей линии (ГНЛ) среди уже размещенных объектов

Для реализации дополнительных возможностей алгоритма A^* , описанных в предыдущем подразделе, внесем следующие изменения в известный алгоритм A^* .

1. **Алгоритм проведения трассы заданного размера.** Для обеспечения возможности проведения трассы переменного размера используем известный прием - использование адаптивной сетки. Для наглядности будем описывать применение этого прием на 2D – сетке. На рисунке 3.9 представлена плоская рецепторная матрица размером 10x10 элементов, где ячейками с цифрой “1” обозначена зона запрета, а ячейками с “0” – свободное пространство. На рисунке 3.9 а показано проведение трассы известными способами шириной в 1 рецептор. Если же нам необходимо построить трассу шириной 2 рецептора, мы искусственно укрупняем рецепторную матрицу, разбивается на дополнительные клетки, каждая из которых по длине в 2 раза больше первоначального рецептора (например, $Yn/2$ и $Xn/2$) – рисунок 3.9 б.

Таким образом, размер сетки станет равным 5x5. Если в новой клетке размером 2x2 первоначальных рецептора есть хотя бы одна ячейка с “1”, то вся эта область считается зоной запрета. В нашем примере

0	0
1	0

 и

1	1
0	1

 теперь полагаются препятствием (рисунок 3.9 в), т.е. произошло переименование клеток. Затем применим алгоритм A^* для нахождения пути между начальной и конечной точкой на сетке размером 5x5. В результате будет построен путь шириной 2 рецептора с учетом препятствий. После этого переложим полученный результат на исходную сетку размером 10x10. Если необходимо учесть безопасную зону между препятствием и трассой (соблюсти δ - окрестность), ее заданные размеры также учитываются добавлением количества рецепторов δ при перерасчете сетки. Сравнение двух путей представлено ниже на рисунке 3.10

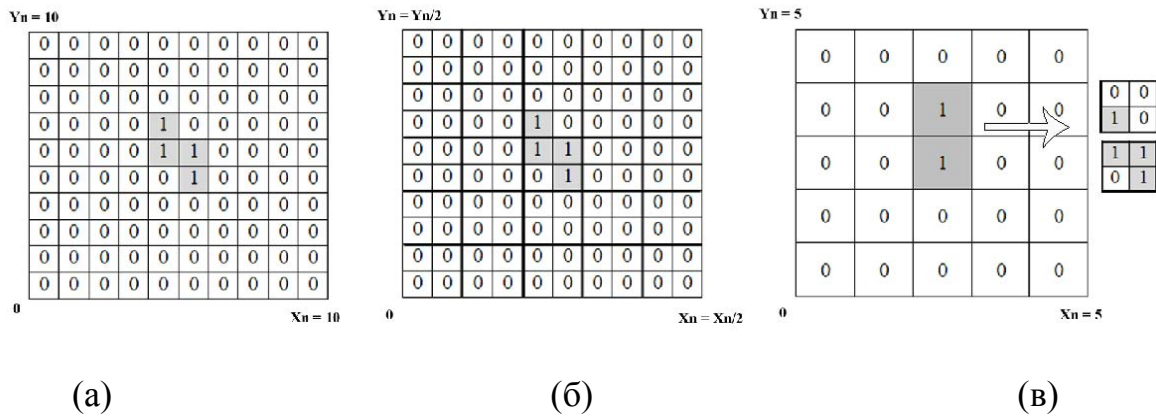
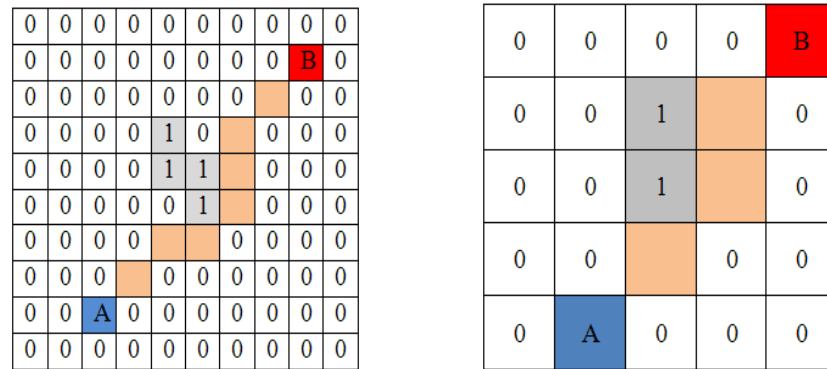


Рисунок 3.9– Плоская (2D) рецепторная матрица с переменным размеров ячеек: *a* – исходная; *б* – укрупнение клеток; *в* – переименование клеток



а) Path = 1px, Map = 10x10 px б) Path = 2px, Map = 5x5 px

Рисунок 3.10 - Изменение рецепторной матрицы при изменении размеров трассы:
a – начальная рецепторная матрица; *б* - конечная

2. Алгоритм построения гладкого пути. Разработанный нами алгоритм трассировки сглаженного пути является глубокой модификацией алгоритма A^* и предусматривает выполнение следующих основных действий:

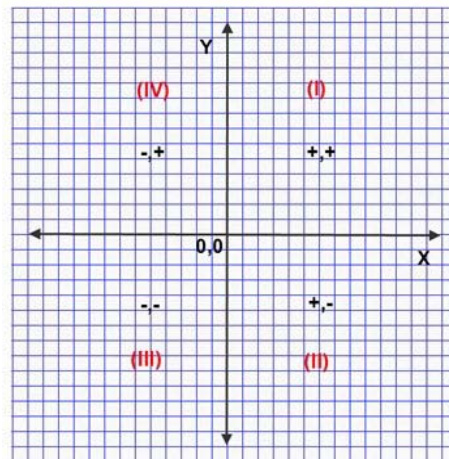
1. Ввод параметров трассы для нахождения рационального пути между двумя исходными точками трассы: начальной *A* и конечной *B* в 2D или 3D рецепторной матрице.

2. Нахождение основной трассы между двумя исходными и конечными точками A и B, используя улучшенный алгоритм A* введя в него дополнительно разработанный нами компонент «Штрафы за смену направления». В этом компоненте "Штрафов за смену направления" стоимость пути возрастает всякий раз, когда путь меняет направление, то есть образуется угол. В результате, если путь найден, он будет более гладким, так как исключаются «зигзаги» и трасса становится более плавной. Недостатком применения этого метода является то, что время расчета увеличивается, так как приходится искать дополнительные узлы).

3. Нахождение всех возможных углов n основного пути (переменная NOA хранит количество углов n).

4. Возьмем из очереди координаты узла $NODE_{i++}$. Будем считать, что $NODE_i := NODE_{i++}$ – угол основного пути.

5. Проверяем, как располагается узел $NODE_i$ на квадратичном графе. В данном алгоритме мы анализируем 8 возможных типов углов: 4 типа для манхэттонского пути и 4 - для диагонального (рисунок 3. 11).



а) Квадратичный граф

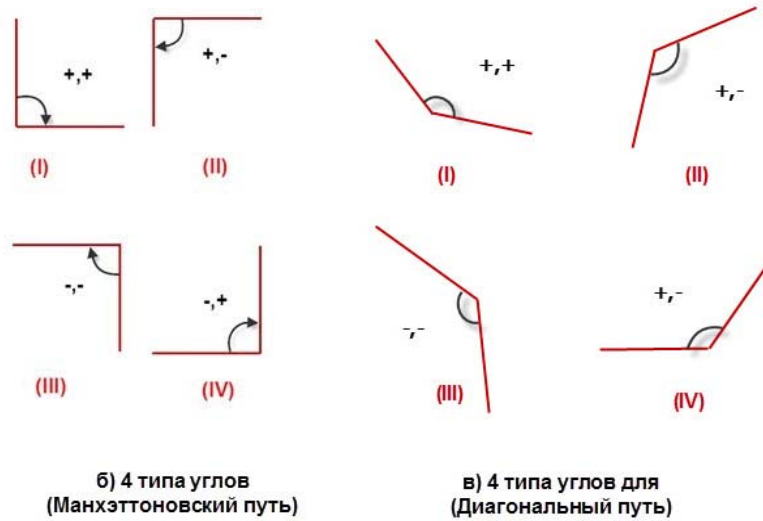


Рисунок 3.11 – Анализируемые в алгоритме восемь типов углов на квадратичном графе

6. Проверяем угол $NODE_i$ на наличие пустого пространства в соответствии с типом угла.

7. Проверяем условие достижения препятствия в $SECTOR_Rmin_{ij}$ (объект или зона запрета), и если оно не выполняется, то берется участок пути $SECTOR_Rmin_{ij}$ как свободное пространство для модификации оригинального пути.

8. Осуществляем передачу координат узла $NODE_i$ и участка пути $SECTOR_Rmin_{ij}$ в подпрограмму (подпроцесс) сглаживания пути для нахождения гладкого пути с использованием алгоритма APC.

9. Повторяем шаги 4 – 10.

Структурная схема разработанного нами алгоритма приведена на рисунке 3.12.

Геометрическая иллюстрация этапов работы алгоритма показана на следующих рисунках. На *этапе 1* производится поиск рационального пути между двумя точками с использованием улучшенного алгоритма A^* , основанный на диагональном или манхэттоновском поиске направления (в примере используется манхэттоновский метод – рисунок 3.13).

На *этапе 2* производится нахождение углов излома трассы и определение их типов на всей длине траектории. На этом же этапе производится разбивка отрезков прямолинейных участков трассы на две равные части (т.е. пополам). На рисунке 3.14 середины этих отрезков помечены засечками.

На *этапе 3* производится как вспомогательное построение вычисление координаты центра окружности скругления, вписанной в соответствующий угол. Центр этой окружности скругления определяется по известным в аналитической геометрии формулам построения окружности по трем заданным точкам. На рисунке 3.15 показаны эти исходные точки - вершина угла P_1 и точки, заданные длинами отрезков X_1 и X_2 , что позволяет аналитически определить положение центра окружности сопряжения – координаты точки C_1 .

На *этапе 4* производится нахождение некоторой точки P'_1 на отрезке P_1C_1 . Это выполняется в несколько шагов:

Шаг1. Определяем отрезок P_1C_1 , соединяющий точки P_1 и C_1 .

Шаг2. Проверяем все точки отрезка P_1C_1 , пока не получим подходящую точку для построения дуги (используется метод построения окружности по трем точкам, при этом учитываются области запрета).

Шаг3. Через полученную точку P'_1 рисуем арку.

Найденная таким образом точка P'_1 используется для построения новой дуги с центром в точке C'_1 , которая также строится (аналитически) по трем точкам – двум старым (серединам отрезков X_1 и X_2 и одной новой – ранее найденной точке P'_1) (рисунок 3.16).

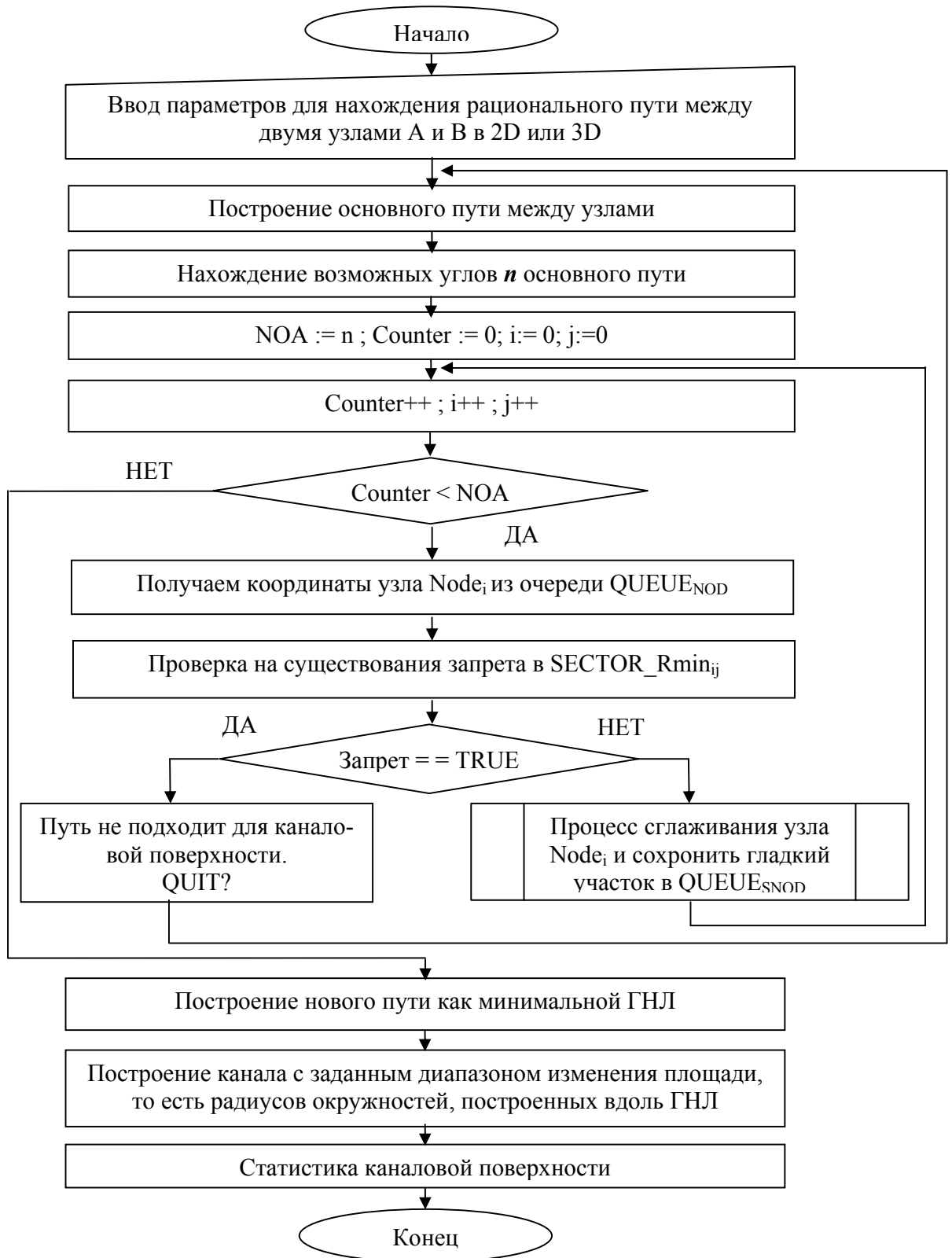


Рисунок 3.12 – Структурная схема алгоритма, обеспечивающего сглаживание трассы

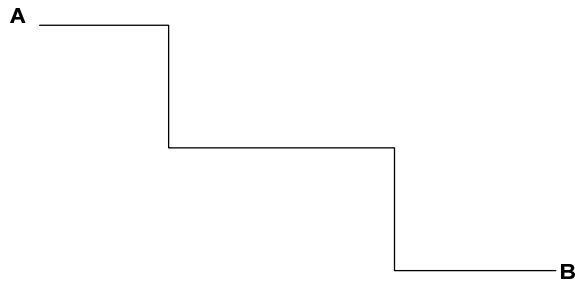


Рисунок 3.13 – Этап 1 - построение между исходной и конечной точками манхэттенской или диагональной трассы

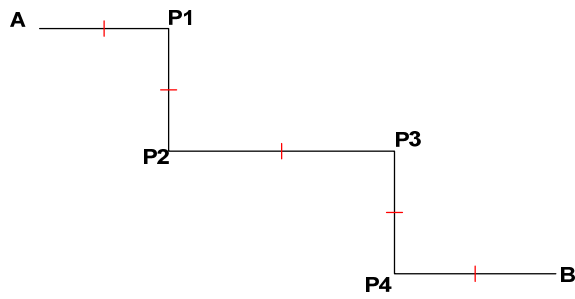


Рисунок 3.14 – Этап 2 – нахождение углов излома трассы и определение их типов на всей длине траектории и разбивка отрезков прямолинейных участков трассы на равные части

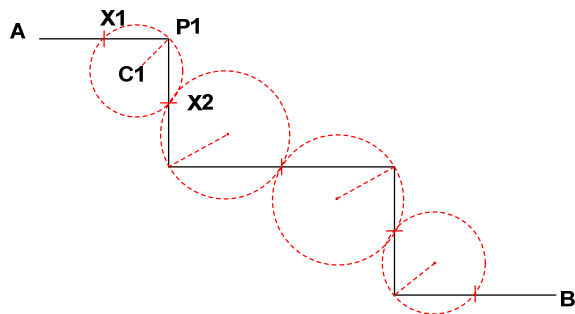


Рисунок 3.15 – Этап 3 – нахождение углов излома трассы и определение их типов

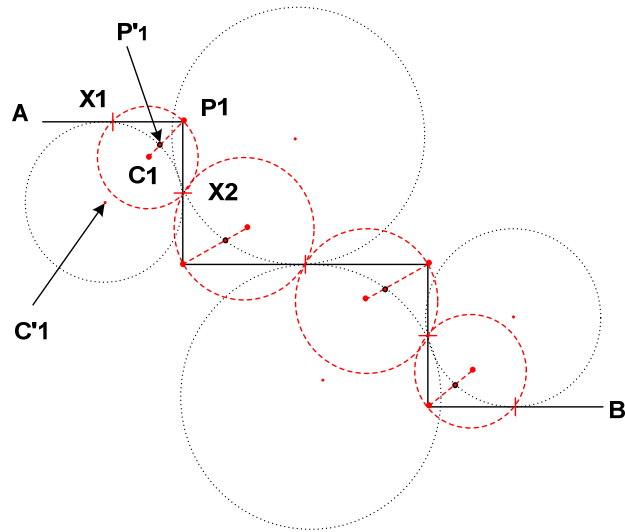


Рисунок 3.16 – Этап 4 – построение новой дуги с центром в точке C'_1 , по трем точкам – двум старым и одной новой – точке P'_1

На *этапе 5* описанная на предыдущем этапе 4 операция повторяется для всех прямолинейных участков первоначальной манхеттеновской трассы. В результате этого трасса, состоящая из отрезков прямых заменяется плавной трассой, образованной сопряжением дуг различных радиусов, лежащих в одной плоскости (рисунок 3. 17).

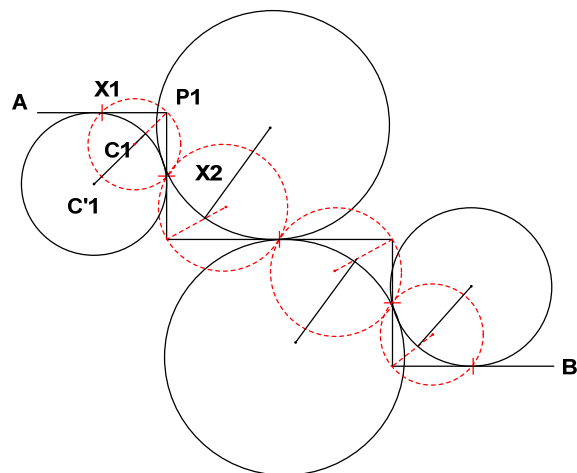


Рисунок 3.17 – Представление прямолинейных отрезков трассы сопряжением дуг различных диаметров

На *этапе 6* мы “окончательно забываем” о ранее спроектированной на этапе 1 манхэттенской трассы и переходим к плавной траектории, полученной на этапе 5 сопряжением дуг различного диаметра. Здесь же происходит изменение информационной структуры алгоритма – массив рецепторов заменяется намного более компактной информацией о совокупности параметров сопряженных дуг, задающих в совокупности новую плавную траекторию, и являющуюся ГНЛ проектируемого канала (рисунок 3.18).

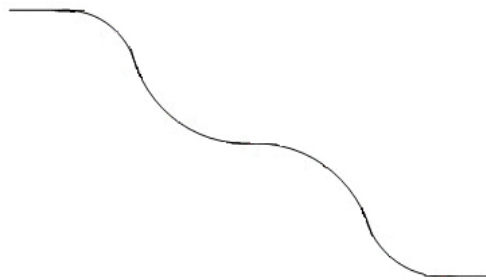


Рисунок 3.18 – Этап 6 – окончательное формирование ГНЛ как плавной траектории

Таким образом, идея разработанного нами алгоритма – построение сначала манхэттенской трассы между заданными исходной и конечной точками, затем выполнение расчетным путем вспомогательных построений на прямолинейных отрезках трассы, затем аппроксимация этой трассы совокупностью дуг максимально возможного радиуса. Если минимальный из этих радиусов не меньше заданного параметра минимального радиуса R_{min} , то построенная траектория считается удовлетворительно. Процесс сглаживания манхэттенской траектории показан на рисунке 3.19.

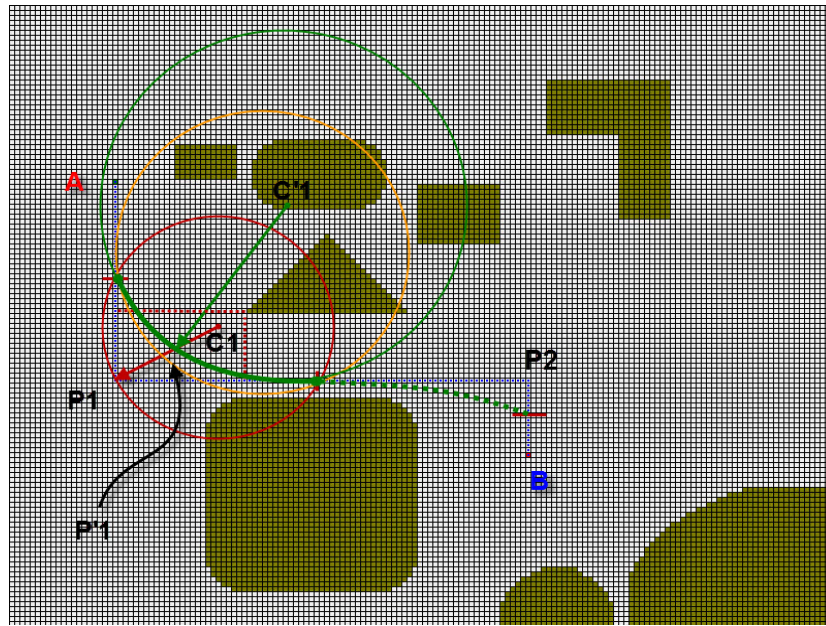


Рисунок 3.19 - Сглаживание манхэттенского пути с учетом заданного R_{min} с использованием нашего алгоритма

На *этапе 7* по результатам работы предыдущих этапов с центром сечений расположенных на ГНЛ происходит формирование каналового тела заданного сечения либо по плоскостям параллелизма (рисунок 3.20 а), либо по плоскостям, расположенным нормально по отношению к ГНЛ (рисунок 3.20 б).

На *этапе 8* происходит проверка условия взаимного непересечения спроектированного канала с областями запрета. Эта процедура производится стандартным пересчетом – проверкой рецепторов канала на пересечение с рецепторами, имеющими значение “1”. Так как эта процедура проверки непересечения является типовой для рецепторных геометрических моделей, она здесь подробно не описывается.

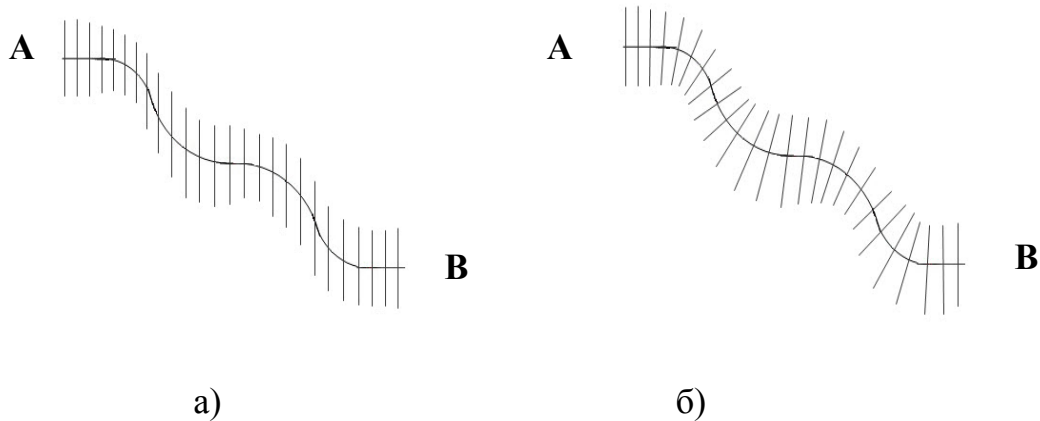


Рисунок 3.20 – Этап 7 – построение каналového тела по ранее сформированной трассе (ГНЛ): а – с плоскостью параллелизма; б – по нормальным сечениям

Вопрос о прохождении трассы на заданном расстоянии от областей запрета решается сравнительно просто. Для обеспечения прохождения трассы вблизи δ -окрестности мы перед началом всех расчетов увеличиваем размер препятствия (области запрета) именно на величину δ -окрестности. Таким образом, препятствие как бы “разбухает” при построении трассы, а затем снова возвращается к своим первоначальным размерам, обеспечивая тем самым заданный зазор при прохождении трассы.

3.4 Разработка эвристик для построения главной направляющей линии канала

Как уже было отмечено ранее, наиболее сильной стороной алгоритма A^* , обусловившей его выбор нами в качестве прототипа - это возможность оптимального и эвристического поиска трассы. В литературе показано, что во многих случаях эвристический поиск работает лучше других стратегий поиска [169].

Эвристическая функция алгоритма $h(n)$ определяет выбор направления поиска к целевой вершине. Если эвристическая функция допустима (то есть не пре-

вышает минимальную стоимость графа к целевой вершине), тогда алгоритм A^* гарантировано найдет кратчайший путь. Однако предпочтительнее использовать эвристику, которая снижает минимальную стоимость настолько, насколько это возможно, что приведет к уменьшению числа вершин, которые необходимо рассмотреть. Идеальная эвристика всегда возвращает фактическую минимальную стоимость, необходимую для достижения оптимальной по расстоянию вершины. Эвристика манхэттенского расстояния, представленная на рисунке 3.21 а, определяется как $h(n) = |x_a - x_b| + |y_a - y_b|$,

где (x_a, y_a) и (x_b, y_b) координаты текущей и целевой вершины, соответственно. Эвристика идеальна, когда используется 4 смежные вершины. Эвклидова эвристика (рисунок 3.21 б) определяется, как $h(n) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$ и является допустимой, но, как правило, значительно превышает фактическую стоимость пути. Это означает, что алгоритм обойдет слишком много лишних вершин, что увеличит время нахождения пути.

Эвристика диагонального расстояния [194] определяется, как $h(n) = |x_a - x_b| + |y_a - y_b| + (\sqrt{2} - 2) \min(|x_a - x_b|, |y_a - y_b|)$ и представлена на рисунке 3.21 в. Эвристику, сочетающую в себе компоненты манхэттенской и евклидовой, можно считать рациональной. Ее преимуществом является то, что определяется фактическая минимально возможная стоимость пути к цели, если используется 8 смежных вершин. Кроме того, в вычислениях удастся избавиться от квадратного корня, что позволяет проводить вычисления значительно быстрее и эффективнее, чем при евклидовой эвристике. Было обнаружено, что использование эвристики манхэттенского расстояния позволяет производить поиск на 40% быстрее, по сравнению с двумя другими эвристиками, при небольшом увеличении стоимости решения, под которым мы понимаем необходимые интегральные затраты вычислительной мощности (производительности про-

цессора, оперативной памяти компьютера и т.д.), влияющие на время компьютерной реализации данной эвристики.

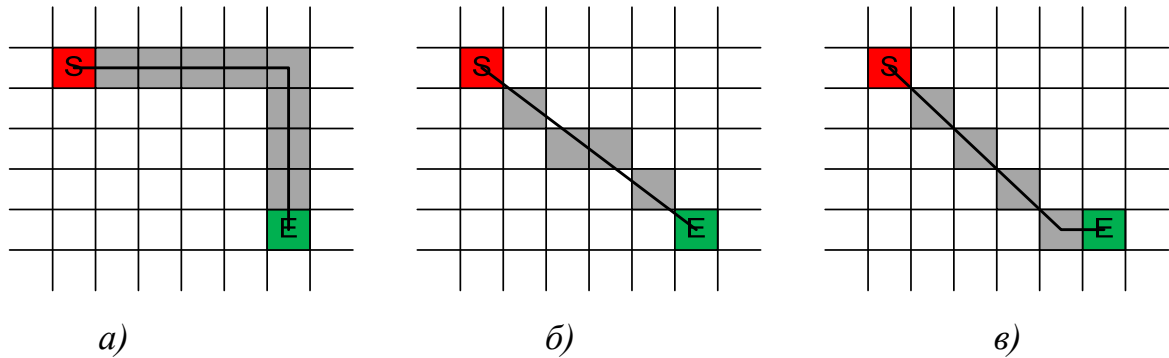


Рисунок 3.21 –Используемые эвристики при Выборе направления к целевой точке: *a* – ориентированные на манхэттенское расстояние; *б* - на евклидово расстояние; *в* - на диагональное расстояние

Другим важнейшим вопросом, определяющим функционирование нашего алгоритма, является **выбор направления поиска**, также определяемый эвристическими методами. Для нахождения пути от начальной до целевой вершины необходимо определить, каким образом будут выбираться дочерние вершины. Обычно алгоритмы поиска пути ведут поиск в 4 и 8 направлениях (рисунок 3.22 *a*, *б*). Но в нашем алгоритме для поиска пути в 3D используется многонаправленный выбор направления, показанный на рисунке 3.22 *в*.

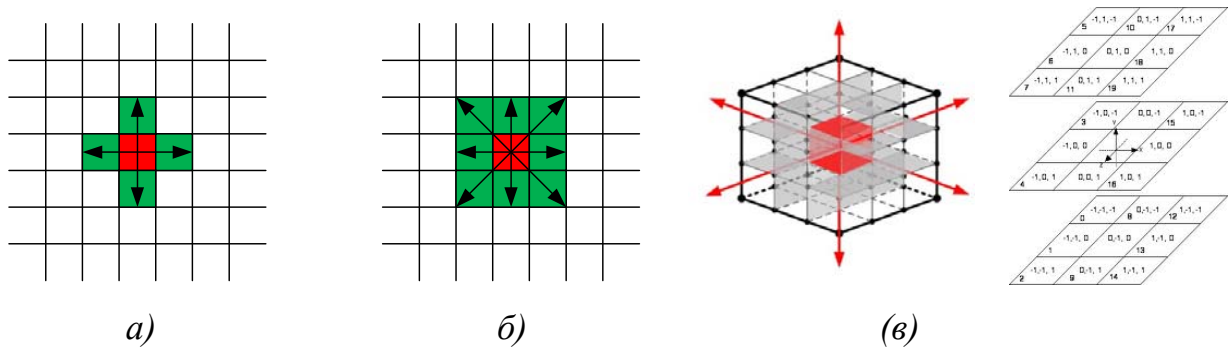


Рисунок 3.22- Различные направления поиска пути: *a* - по 4-м смежным вершинам; *б* – по 8-ми смежным вершинам; *в* – по 26-и смежным вершинам

Иллюстрация этих направлений поиска в рецепторной матрице в пространственном изображении приведена на рисунке 3.23.

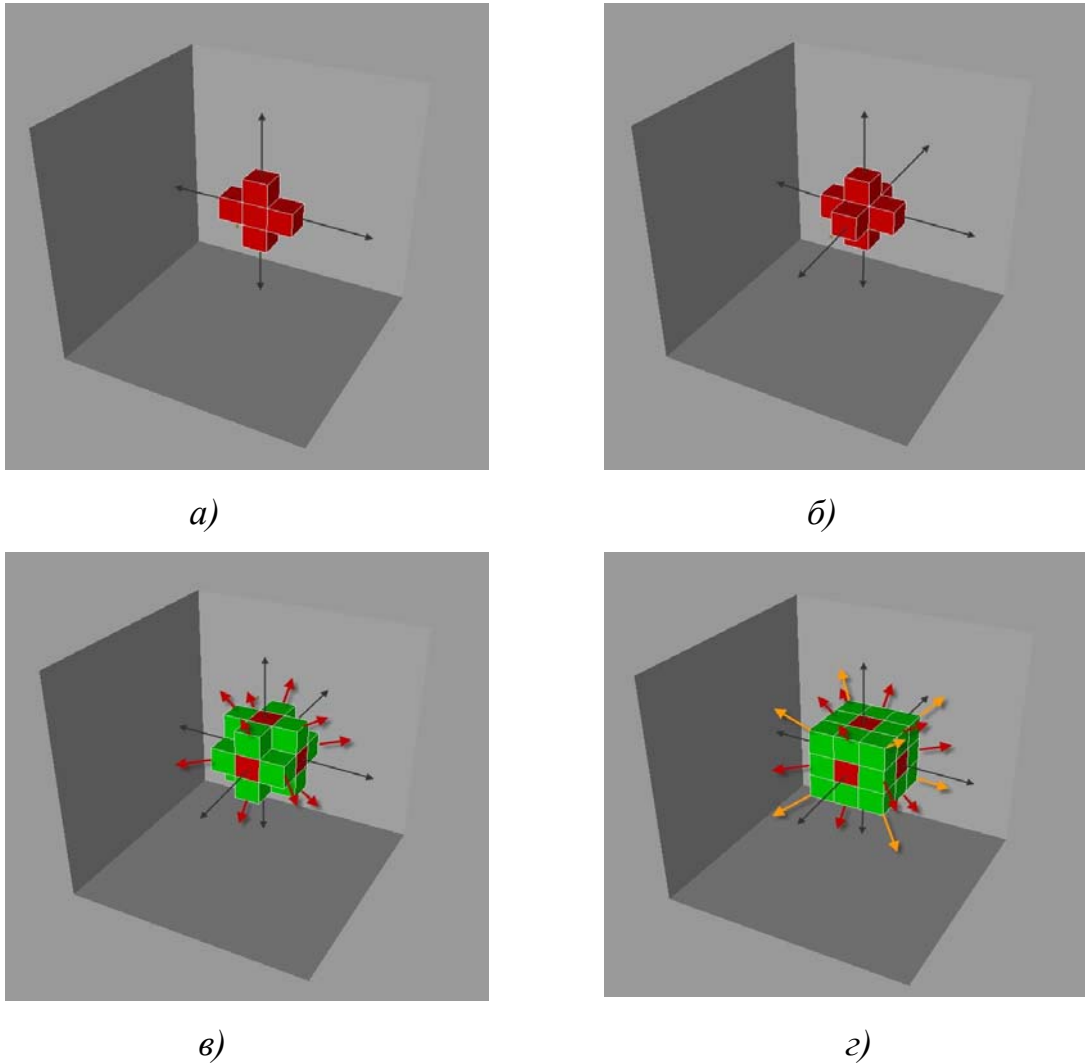


Рисунок 3.23 - Поиск пути в рецепторном поле в 3D изображении: *a* - по 4-м смежным вершинам; *б* – по 8-ми смежным вершинам; *в* – по 26-и смежным вершинам

В разработанном нами алгоритме используется эвристика, предусматривающая многонаправленный поиск по 8-ми направлениям, если построение канала происходит в плоскости (хотя и в 3D- рецепторной матрице – рисунок 3.24), и по 26-и смежным вершинам при проектировании пространственного канала.

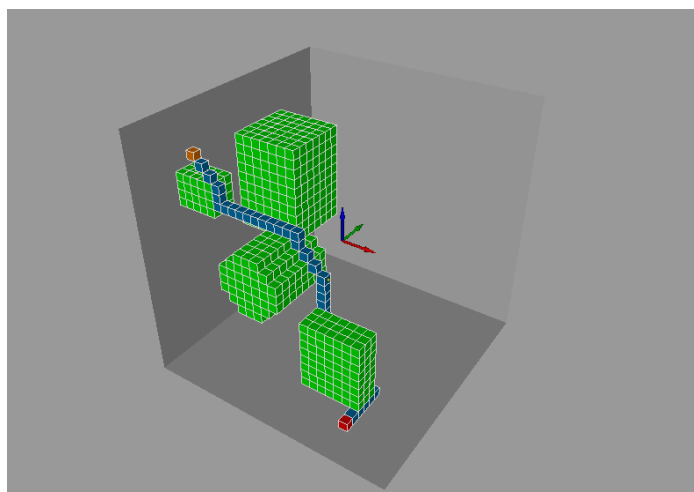


Рисунок 3.24 - Нахождение пути в 3D рецепторной модели

Существенной проблемой при проектировании каналовой поверхности рецепторным методом является переход от прохождения главной направляющей линии (ГНЛ) в плоскости (как на приведенном ранее рисунке 3.24) к пространственной ГНЛ. Особенно сложной эта проблема становится при построении канала по нормальным сечениям (см. рисунок 3.20 б). В этом случае за дистанцию, относительно которой изменяется закон площадей сечений, принимается криволинейная ось s – средняя линия тока жидкости или газа, которая в последнем случае является сложной пространственной кривой в системе координат $Oxyz$ по длине ГНЛ – параметру P (см. рисунок 1.16):

$$x_s = x_s(P)$$

$$y_s = y_s(P)$$

$$z_s = z_s(P)$$

В соответствии с классической дифференциальной геометрией на рисунке 1.16 видно, что по оси s , как по направляющей, перемещается начало локальной системы координат $Ox^*y^*z^*$. Угловые положения осей x^* , y^* и z^* в зависимости от значения параметры длины P по ГНЛ определяют нормальную плоскость сечения и определяются функциями направляющих косинусов:

$$\cos \alpha_l(P), \quad \cos \beta_l(P), \quad \cos \gamma_l(P),$$

$$\begin{aligned} & \cos \alpha_2 (P), \quad \cos \beta_2 (P), \quad \cos \gamma_2 (P), \\ & \cos \alpha_3 (P), \quad \cos \beta_3 (P), \quad \cos \gamma_3 (P) \end{aligned}$$

определяющих в каждой точке O^* значение величин кривизны κ_O и кручения τ_O . В соответствии с известными формулами дифференциальной геометрии направляющими косинусами определяется положение векторов касательной \vec{t} $[\cos \alpha_1 (P), \cos \beta_1 (P), \cos \gamma_1 (P)]$, главной нормали \vec{n} $[\cos \alpha_2 (P), \cos \beta_2 (P), \cos \gamma_2 (P)]$ и бинормали \vec{b} $[\cos \alpha_3 (P), \cos \beta_3 (P), \cos \gamma_3 (P)]$ решением системы уравнений:

$$\begin{cases} \cos \alpha_1 \cdot \cos \alpha_2 + \cos \beta_1 \cdot \cos \beta_2 + \cos \gamma_1 \cdot \cos \gamma_2 = 0 \\ \cos \alpha_1 \cdot \cos \alpha_3 + \cos \beta_1 \cdot \cos \beta_3 + \cos \gamma_1 \cdot \cos \gamma_3 = 0 \\ \cos \alpha_2 \cdot \cos \alpha_3 + \cos \beta_2 \cdot \cos \beta_3 + \cos \gamma_2 \cdot \cos \gamma_3 = 0 \\ \cos \alpha_1^2 + \cos \beta_1^2 + \cos \gamma_1^2 = 1 \\ \cos \alpha_2^2 + \cos \beta_2^2 + \cos \gamma_2^2 = 1 \\ \cos \alpha_3^2 + \cos \beta_3^2 + \cos \gamma_3^2 = 1 \end{cases}$$

Из этой системы уравнений видно, что расчет направляющих косинусов векторов трехгранника Френе по этой системе уравнений представляет собой достаточно громоздкую вычислительную процедуру. Кроме того, в исследовании Андреева В.А. [5] показано, что способ формирования поперечных сечений в системе векторов трехгранника Френе, помимо вычислительных сложностей, имеет существенный недостаток для практики. При движении подвижного трехгранника Френе вдоль пространственной ГНЛ он вращается вокруг мгновенной оси, направление которой определено вектором Дарбу с некоторой угловой скоростью. Поэтому каждое следующее сечение вдоль ГНЛ поворачивается по отношению к предыдущему сечению на некоторый угол, в результате чего канал получает некоторую нежелательную закрутку.

Поэтому для расчета положения нормальной плоскости по отношению к ГНЛ использован подход, предложенный в диссертации Андреева В.А. [5], основанный на несколько более простых формулах:

1. Вычисляются первые и вторые производные по параметру длины P вдоль ГНЛ s : $x', x'', y', y'', z', z''$.

2. Вычисляется кривизна K по формуле:

$$K = \sqrt{x''^2 + y''^2 + z''^2}.$$

3. Определяются направляющие косинусы касательной t^- :

$$\cos \alpha_1 = x'; \cos \beta_1 = y'; \cos \gamma_1 = z'.$$

4. Определяются направляющие косинусы главной нормали n^- :

$$\cos \alpha_2 = \frac{1}{K} x''; \cos \beta_2 = \frac{1}{K} y''; \cos \gamma_2 = \frac{1}{K} z''.$$

Вычисленные таким образом направляющие косинусы позволяют определить положение плоскостей ГНЛ прохождения канала. Эта задача несколько упрощается тем, что, как отмечалось ранее, алгоритм построения 3D трассы основан на чередовании прямых и дуг, расположенных в плоскостях, положение которых в пространстве вычисляется ранее описанными методами (рисунок 3.25).

Если положение вычисленное положение расчетной Φ_1 плоскости совпадает с направлением координатных осей (рисунок 3.26 а), то вышеописанные алгоритмы используются в том виде, в котором они описаны выше. Размеры плоских рецепторов (в сечении) $a \times b$ (рисунок 3.26 б) однозначно определены параметрами рецепторной матрицы $a \times b \times c$ (как правило $a \times a \times a$). Если же расчетная плоскость Φ_2 по отношению к координатным плоскостям занимает общее положение, то плоская рецепторная матрица все равно остается ортогональной, но размеры рецепторов $p \times q$ становятся “динамическими”, определяемыми положением расчетной плоскости добавлением коэффициентов растяжения рецепторов k_1 и k_2 ($p = k_1 \times a$; $q = k_2 \times b$ – рисунок 3.26 в). Таким образом, трехмерная расчетная модель сводится к кусочно-двумерной, если при расчете вместе с положением расчетных

плоскостей одновременно по элементарным формулам вычислять коэффициенты растяжения рецепторов k_1 и k_2 и далее использовать в расчете новые, скорректированные размеры плоских рецепторов.

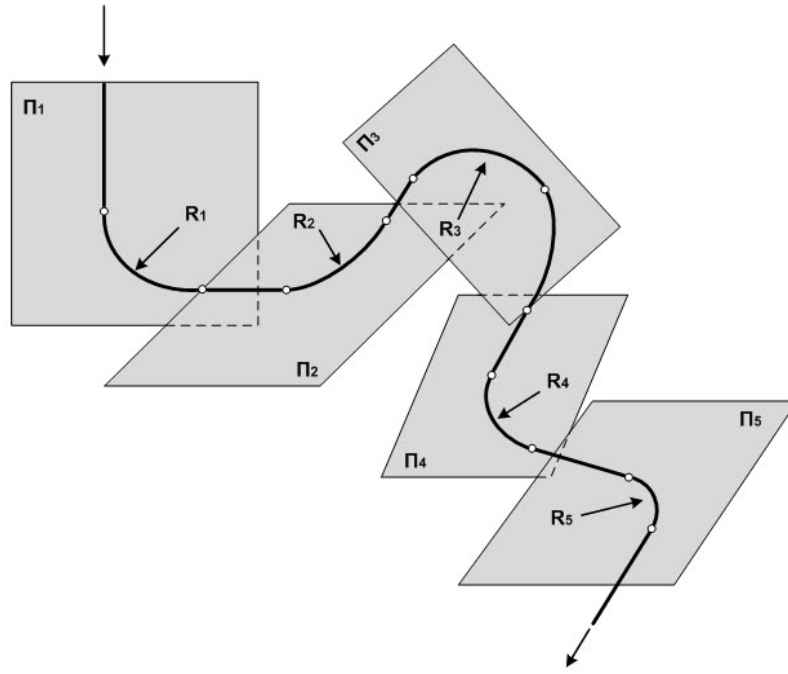
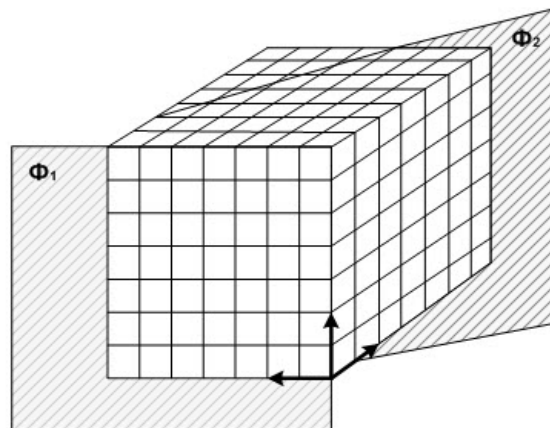


Рисунок 3.25 – Система плоскостей, по которым рассчитывается прохождение ГНЛ канала



a)

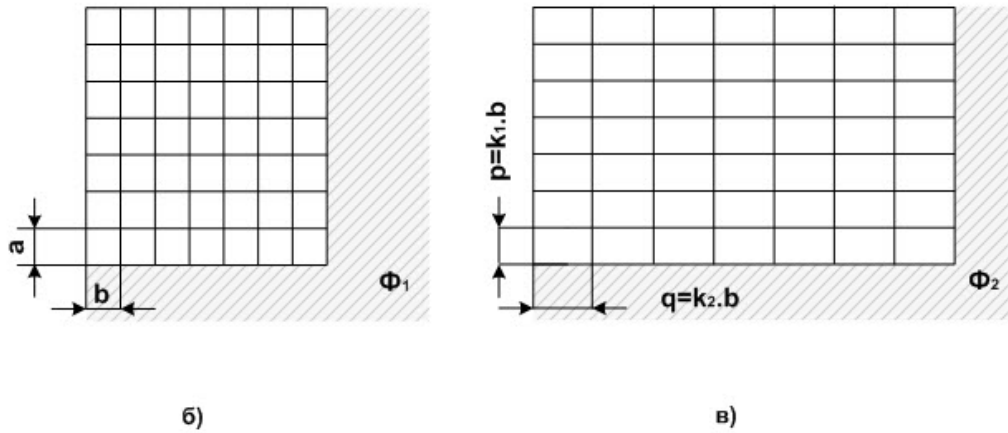


Рисунок 3.26 – Геометрическая иллюстрация сведения трехмерной расчетной модели к двумерной: a – положения расчетных плоскостей Φ_1 (координатной) и Φ_2 (общего положения); b – плоская расчетная рецепторная модель для координатной плоскости; в – плоская расчетная рецепторная модель для плоскости общего положения

ЗАКЛЮЧЕНИЕ И ВЫВОДЫ ПО РАЗДЕЛУ 3

Проведенный анализ научной и технической литературы показал, что задачи плоской и пространственной трассировки являются актуальными для многих отрасле техники и для их реализации разработаны алгоритмы как направленной, так и ненаправленной трассировки. Подробный анализ этих алгоритмов позволил сделать следующие выводы:

1. Все известные алгоритмы трассировки основаны на дискретных (т.е. по существу рецепторных) геометрических моделях.
2. Алгоритмы, реализующие подход направленной трассировки работают намного эффективнее алгоритмов ненаправленной трассировки (даже модифицированных).

3. Проведенный анализ алгоритмов направленной трассировки – алгоритмов Дейкстры, “Лучший-первый” и A^* (A -звездочка) позволил выявить среди них наиболее эффективный, каким оказался алгоритм A^* .

4. Выявлены ограничения использования алгоритма трассировки A^* , не позволяющие использовать его напрямую для автоматизированного проектирования телесной трассировки, наиболее существенными из которых для реализации поставленной задачи является невозможность прокладки трассы на заданном расстоянии от областей запрета и невозможность проектирования плавных трасс.

5. Разработана геометрическая модель и алгоритм построения главной направляющей линии каналовой поверхности для плоской и пространственной трассы, являющийся глубокой модернизацией алгоритма A^* и устраняющий основные ограничения прототипа A^* - возможность прокладки плавных трасс на заданных расстояниях от областей запрета.

6. Разработаны эвристики, повышающие эффективность работы алгоритма трассировки, направленные на выбор рациональных направлений движений к следующей точке будущей траектории.

7. Разработаны алгоритмы сведения трехмерной расчетной задачи к кусочно-двумерной, существенной упрощающей расчетный алгоритм построения ГНЛ канала.

4 АЛГОРИТМИЧЕСКАЯ И ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПРЕДЛОЖЕННОЙ ГЕОМЕТРИЧЕСКОЙ МОДЕЛИ ТЕЛЕСНОЙ ТРАССИРОВКИ

4.1 Алгоритмическая реализация предложенной геометрической модели телесной трассировки

Несмотря на то, что определение условия взаимного непересечения в рецепторных геометрических моделях отличается относительной простотой, программный комплекс, реализующих вышеописанный алгоритм телесной трассировки, является достаточно сложным. Основной алгоритм проектирования трассы был приведен в главе 3 (см. рисунок 3.12). Однако элементы алгоритма, структурная схема которого была приведена на этом рисунке, содержат в себе большое количество дополнительных вычислительных операций, пояснение которых приведено на рисунке 4.1.

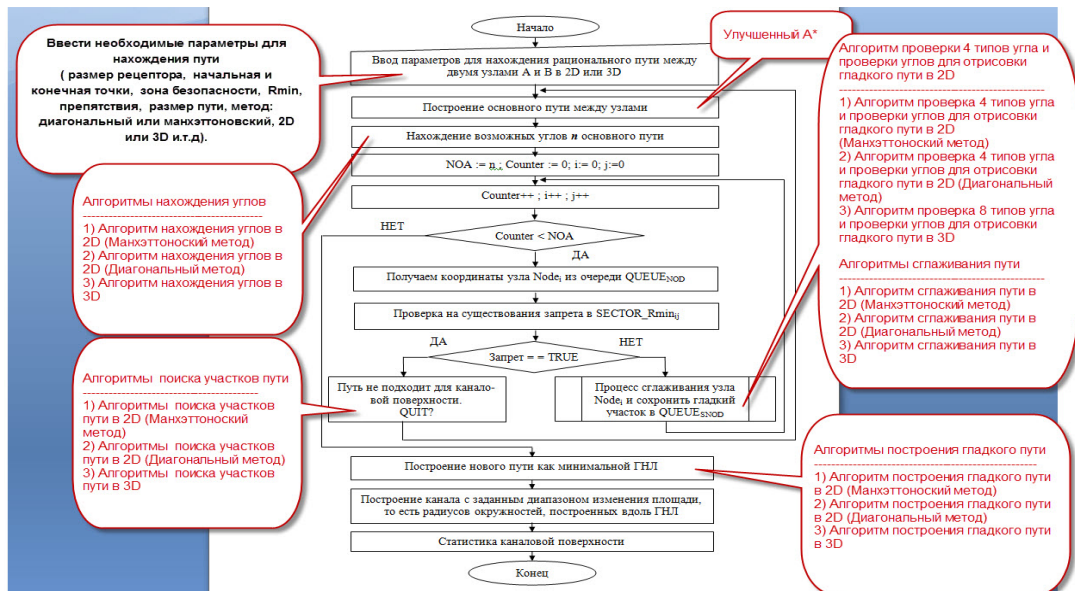


Рисунок 4.1 – Пояснение структуры компонентов основного алгоритма проектирования трассы

Укрупненно последовательность работы алгоритмической структуры предложенного метода геометрического моделирования каналовой поверхности рецепторным методом можно представить в виде следующей последовательности операций:

1. Алгоритмы нахождения углов:

- 1) Алгоритм нахождения углов в 2D (Манхэттенский метод)
- 2) Алгоритм нахождения углов в 2D (Диагональный метод)
- 3) Алгоритм нахождения углов в 3D

2. Алгоритм проверки типов угла и проверки углов для формирования гладкого пути в 2D

- 1) Алгоритм проверка 4 типов угла и проверки углов для формирования гладкого пути в 2D (Манхэттенский метод)
- 2) Алгоритм проверка 4 типов угла и проверки углов для формирования гладкого пути в 2D (Диагональный метод)
- 3) Алгоритм проверка 8 типов угла и проверки углов для формирования гладкого пути в 3D

3. Алгоритмы поиска участков пути

- 1) Алгоритмы поиска участков пути в 2D (Манхэттенский метод)
- 2) Алгоритмы поиска участков пути в 2D (Диагональный метод)
- 3) Алгоритмы поиска участков пути в 3D

4. Алгоритмы сглаживания пути

- 1) Алгоритм сглаживания пути в 2D (Манхэттенский метод)
- 2) Алгоритм сглаживания пути в 2D (Диагональный метод)
- 3) Алгоритм сглаживания пути в 3D

5. Алгоритмы построения гладкого пути

- 1) Алгоритм построения гладкого пути в 2D (Манхэттенский метод)
- 2) Алгоритм построения гладкого пути в 2D (Диагональный метод)
- 3) Алгоритм построения гладкого пути в 3D

6. Структурная схема алгоритма построения гладкого пути

Приведенная на рисунке 4.2 структурная схема алгоритма показывает общие процессы сглаживания пути как в 2D, так и 3D варианте прокладки трассы.

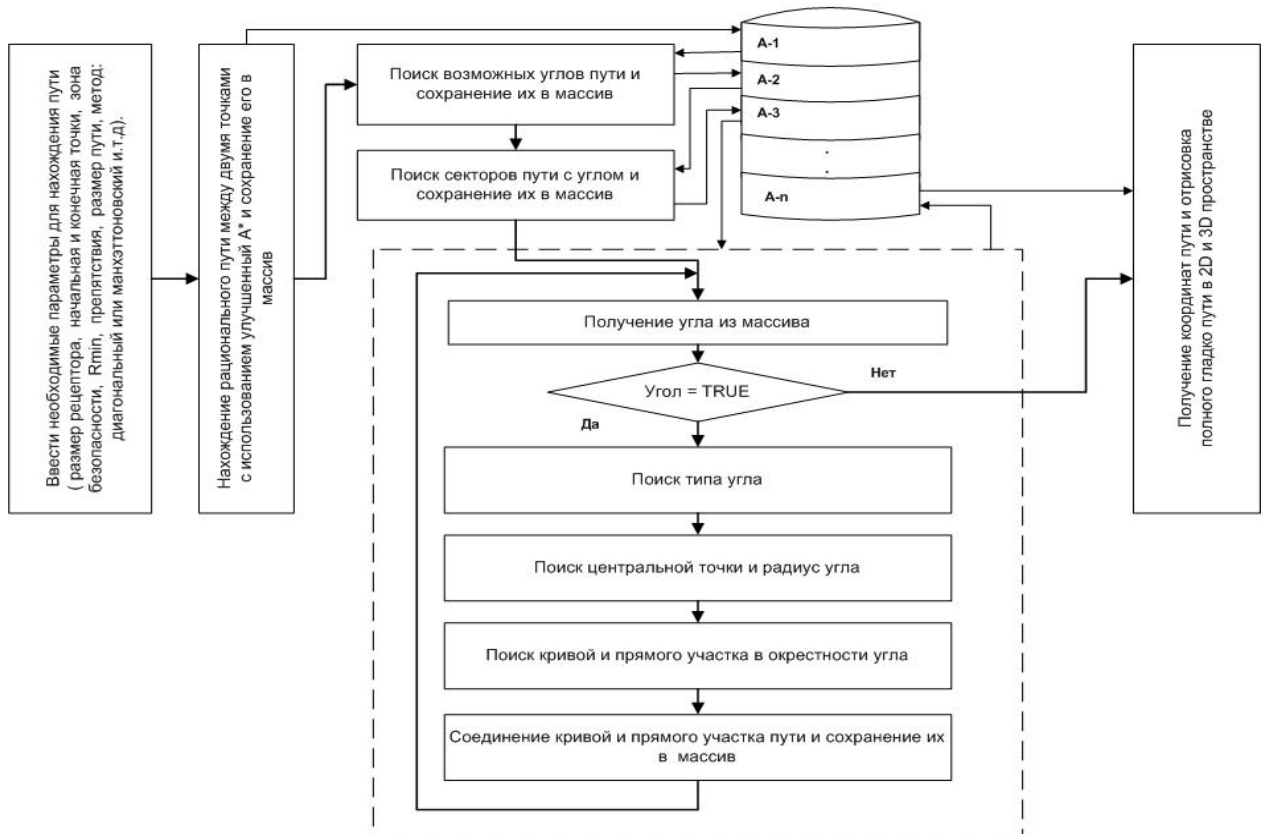


Рисунок 4.2 – Структурная схема алгоритма построения гладкого пути

На рисунке 4.3 показана структурная схема алгоритма процесса (подпрограммы) сглаживания пути в 2D (Манхэттенский метод). Порядок вызова этой подпрограммы *Главная программа* → *Нахождение углов (AnglesIndex)*. Назначение главной программы (основанной на Манхэттенском методе в 2D пространстве):

1) Проверка длины пути, полученного с помощью улучшенного алгоритма A* в 2D пространстве. Если длина пути больше начального значения, заданного главной программой, переходим к процессу, описанному Алгоритмом (1).

2) Проверка углов для последующего сглаживания пути.

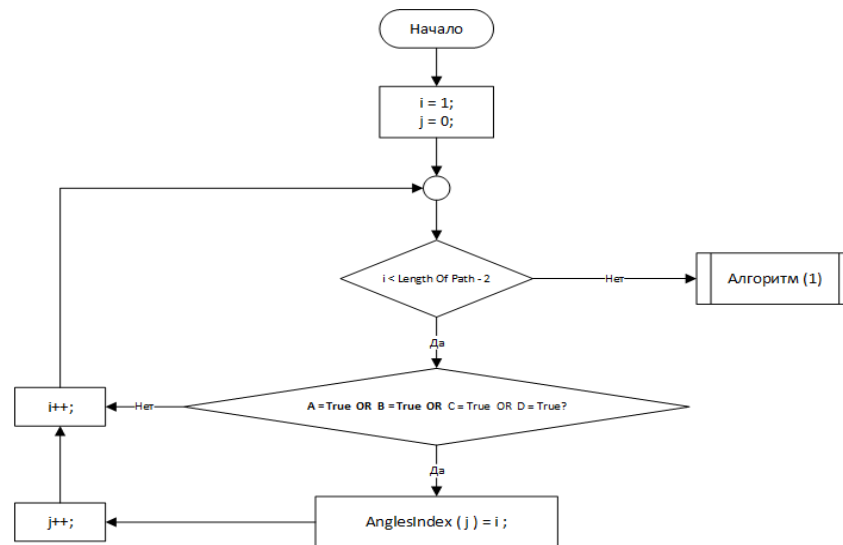


Рисунок 4.3 – Главная программа сглаживания пути в 2D (Манхэттенский метод)

На рисунке 4.4 показана структурная схема алгоритма процесса (подпрограммы) поиска прямолинейных участков пути. Порядок вызова этой подпрограммы **Алгоритм (1) → Поиск участков пути (PartsPath)**. Цели Алгоритма (1) :

- 1) Поиск прямолинейных участков пути в 2D.
- 2) Определение начальной и конечной точки участка пути.

На рисунке 4.5 показана структурная схема алгоритма процесса (подпрограммы) поиска прямолинейных участков пути. Порядок вызова этой подпрограммы **Алгоритма (2) → Расчет для сглаживания каждого участка пути (PartsPath)**. Назначение Алгоритма (2) :

- 1) Проверка количества участков пути.
- 2) Если количество участков (PartPath) больше начального значения, заданного в программе, переход к процессу, описанному Алгоритмом (3). Иначе, переход к Алгоритму (7).

- 1) Поиск прямолинейных участков пути в 2D.

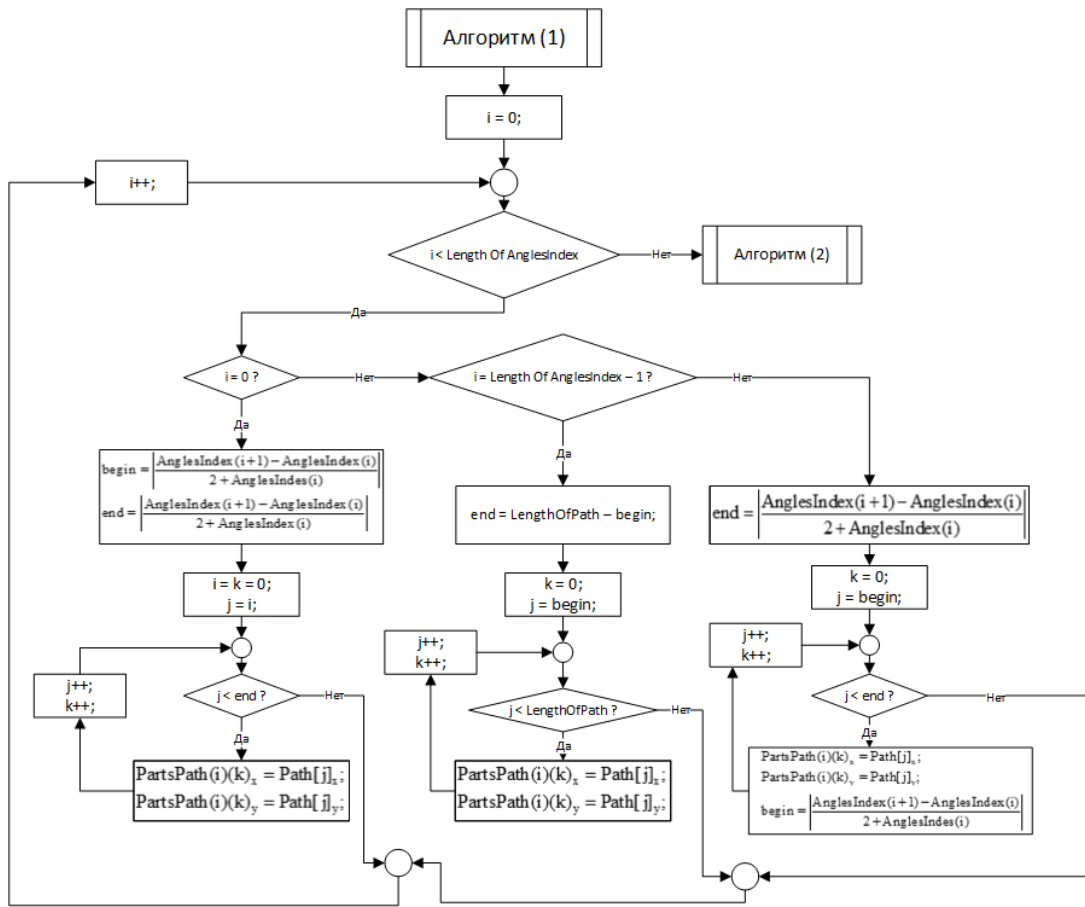


Рисунок 4.4 – Алгоритмы поиска участков пути в плоскости

Алгоритм (2) проверяет 4 типов углов в плоскости, образованными прямыми при предварительном проектировании трассы по Манхэттенской метрике. Тип угла определяется по приведенным ниже формулам.

$$A = \begin{cases} Path(i)_x = Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ \text{AND} \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases}$$

$$B = \begin{cases} Path(i)_x = Path(i-1)_x \text{ and } Path(i)_x > Path(i+1)_x \\ \text{AND} \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases}$$

$$C = \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x = Path(i+1)_x \\ \text{AND} \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y > Path(i+1)_y \end{cases}$$

граммы *Алгоритм(3) → Нахождение типа углов (AngleTypes)*. Назначение Алгоритма (3) :

- 1) Определение тип угла.
- 2) Если текущий угол принадлежит одному из 4 типов, записываются необходимые данные (а именно, начальная, конечная и центральная точки) для формирования окружности, вписанной в текущий угол.

На рисунке 4.7 показана структурная схема алгоритма процесса (подпрограммы) определения типа углов участков пути. Порядок вызова этой подпрограммы *Алгоритм (4) → Нахождение радиуса скругления для каждого угла*. С помощью алгоритма (4) находится радиус и центральная точка окружности, вписанной в текущий угол для его скругления.

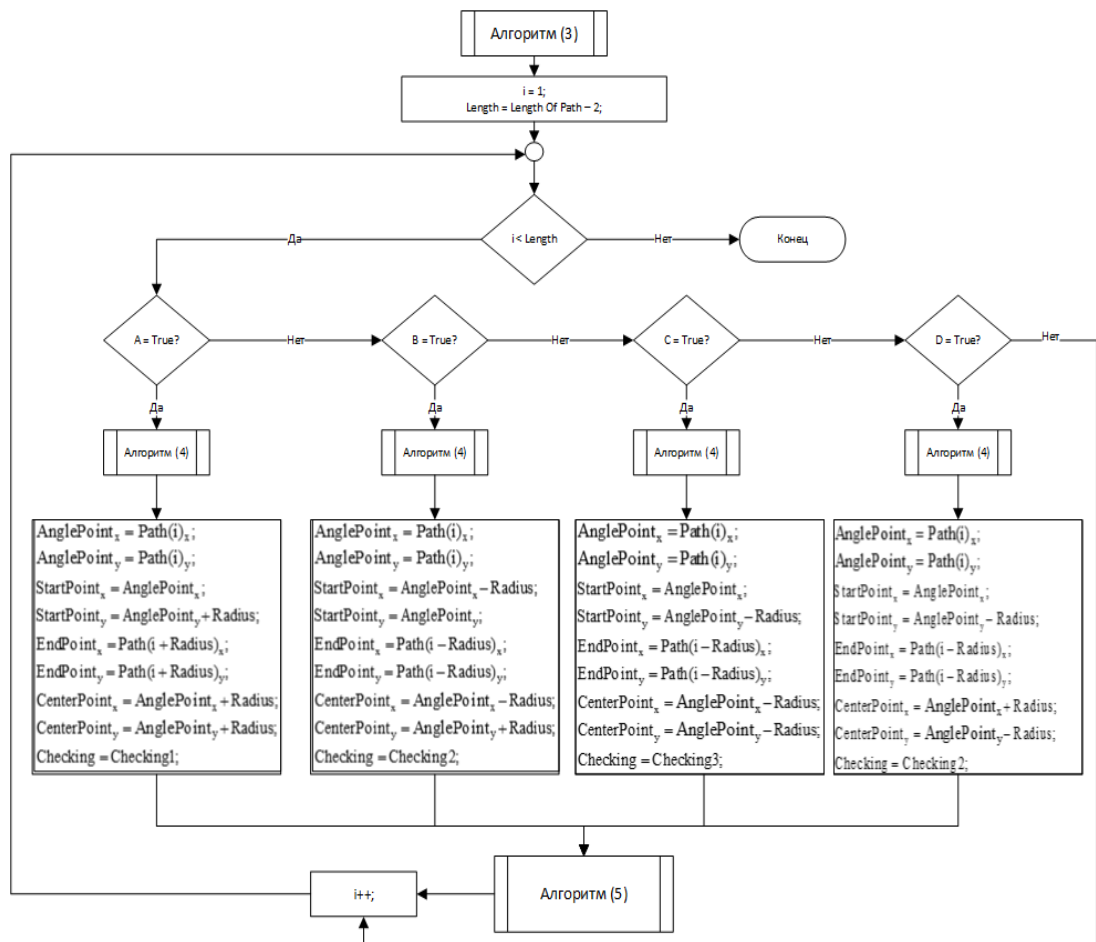


Рисунок 4.6 – Определеие типов углов

На рисунке 4.8 показана структурная схема алгоритма процесса (подпрограммы) определения типа углов участков пути. Порядок вызова этой подпрограммы *Алгоритм (5) → Определение параметров дуги и прокладка трассы по скруглениям рассчитанного радиуса (NewPath)* для каждого угла. Алгоритм (5) используется для построения окружности по известным формулам аналитической геометрии. При этом отображается лишь необходимый для сглаживания пути участком данной окружности (NewPath).

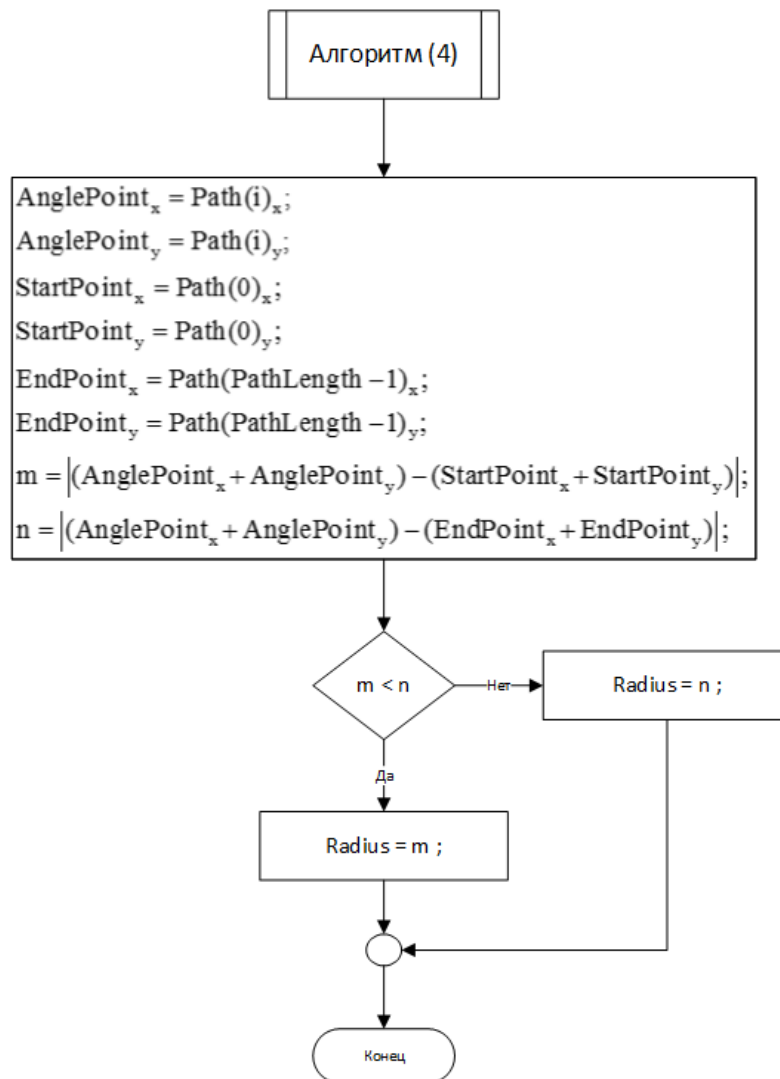


Рисунок 4.7 – Нахождение радиуса и центральной точки окружности скругления прямых

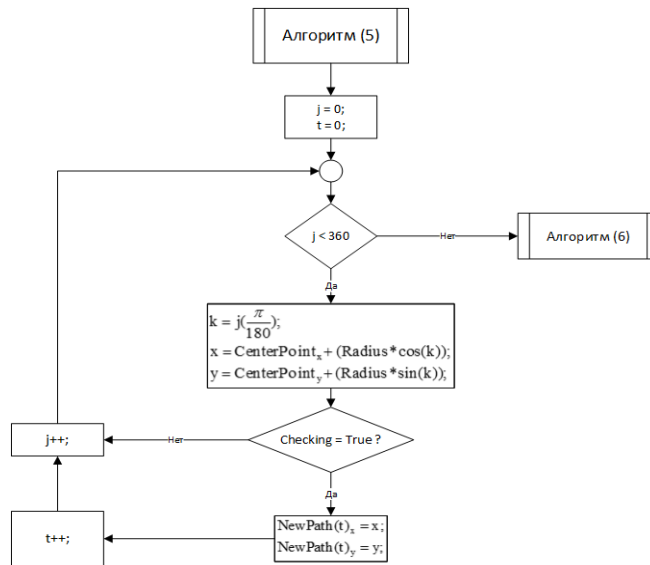


Рисунок 4.8 – Алгоритм определения параметров скругляющей трассу дуги

На рисунке 4.9 показана структурная схема алгоритма процесса (подпрограммы) определения типа углов участков пути. Порядок вызова этой подпрограммы *Алгоритм (6)* → *Получение нормального гладкого пути (NewPath)* для *каждого угла*. Алгоритм (6) находит прямой участок пути для каждого угла в текущем секторе пути в 2D.

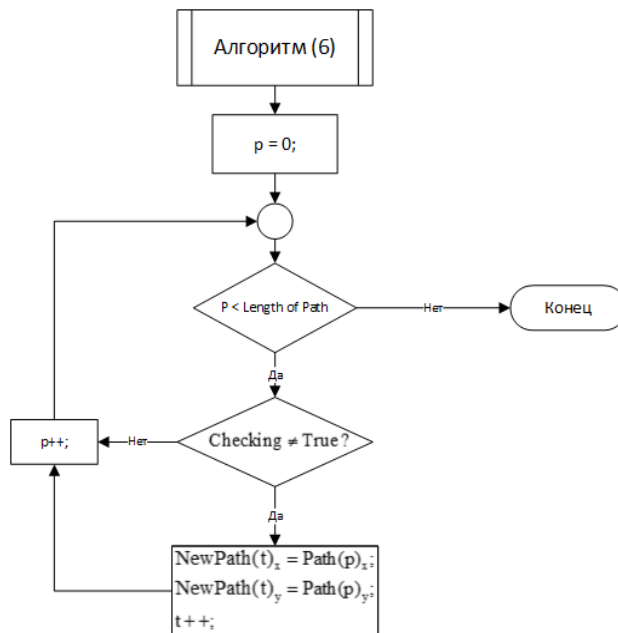


Рисунок 4.9 – Поиск прямого участка пути для каждого угла в текущем секторе

На рисунке 4.10 показана структурная схема алгоритма процесса (подпрограммы) определения типа углов участков пути. Порядок вызова этой подпрограммы *Алгоритм (7) → Формирование дуги и прямой линии для каждого угла, позволяющего получить плавную траекторию*. Целью алгоритма (7) является формирование нового участка гладкой траектории в некоторой промежуточной рабочей плоскости.

На рисунке 4.11 показана структурная схема алгоритма процесса (подпрограммы) процесса сглаживания пути в рабочей плоскости промежуточного положения диагональным методом. Порядок вызова этой подпрограммы *Главная программа → Нахождение углов (AnglesIndex)*. С помощью главной программы (Диагональный метод в плоскости) решаются следующие задачи:

1) Проверка длины пути, полученного с помощью улучшенного алгоритма A* в 2D пространстве. Если длина пути больше начального значения, заданного главной программой, переходим к процессу, описанному алгоритмом (1).

2) Проверка углов для последующего сглаживания пути.

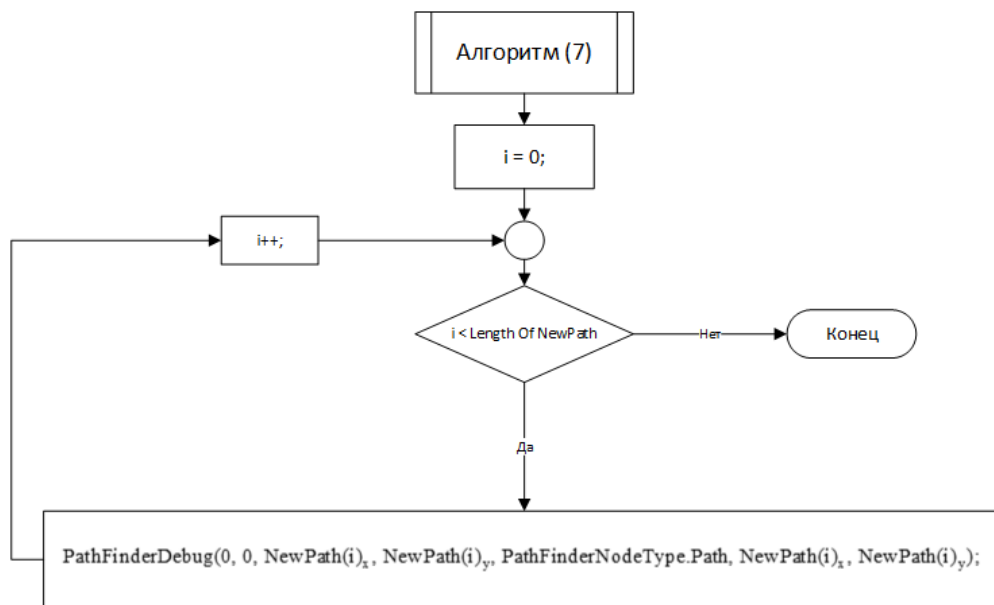


Рисунок 4.10 – Формирование нового участка гладкой траектории в некоторой промежуточной рабочей плоскости

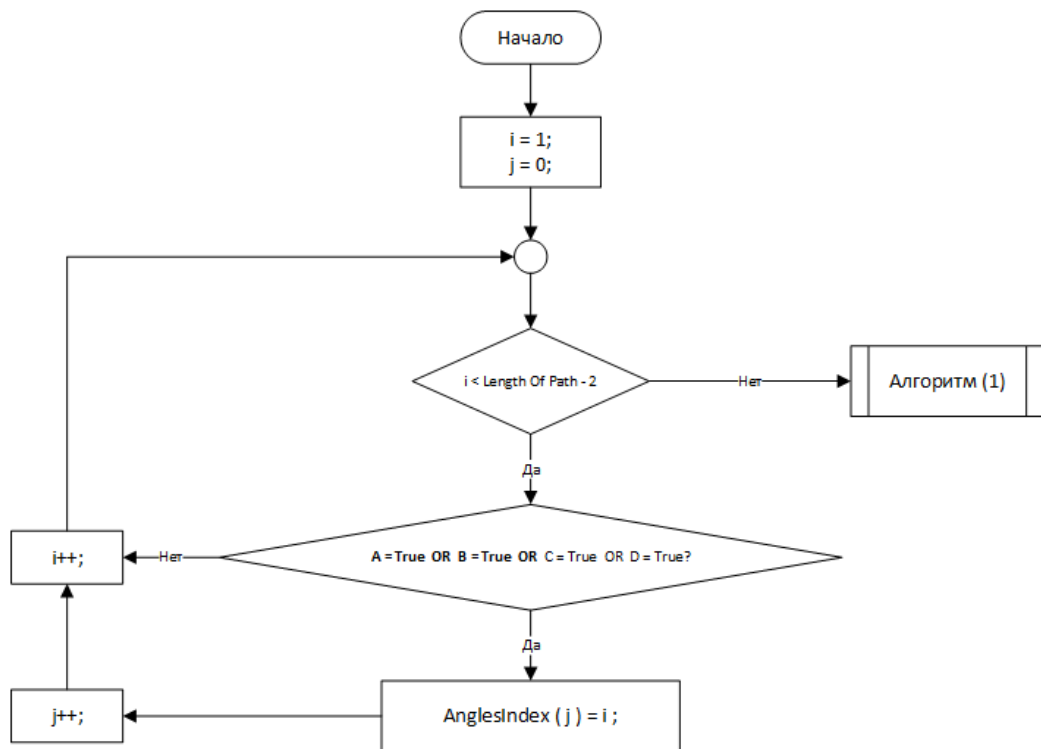


Рисунок 4.11 – Главная программа сглаживания пути в 2D диагональным методом

На рисунке 4.12 показана структурная схема алгоритма процесса (подпрограммы) процесса поиска прямолинейных участков пути в рабочей плоскости промежуточного положения диагональным методом. Порядок вызова этой подпрограммы *Алгоритм (1)* → *Поиск участков пути (PartsPath)*. Обращением к этой подпрограмме определяются следующие данные, необходимые для работы Алгоритма (1):

- 1) Поиск прямолинейных участков пути в плоскости промежуточного положения.
- 2) Определение начальной и конечной точки прямолинейного участка траектории.

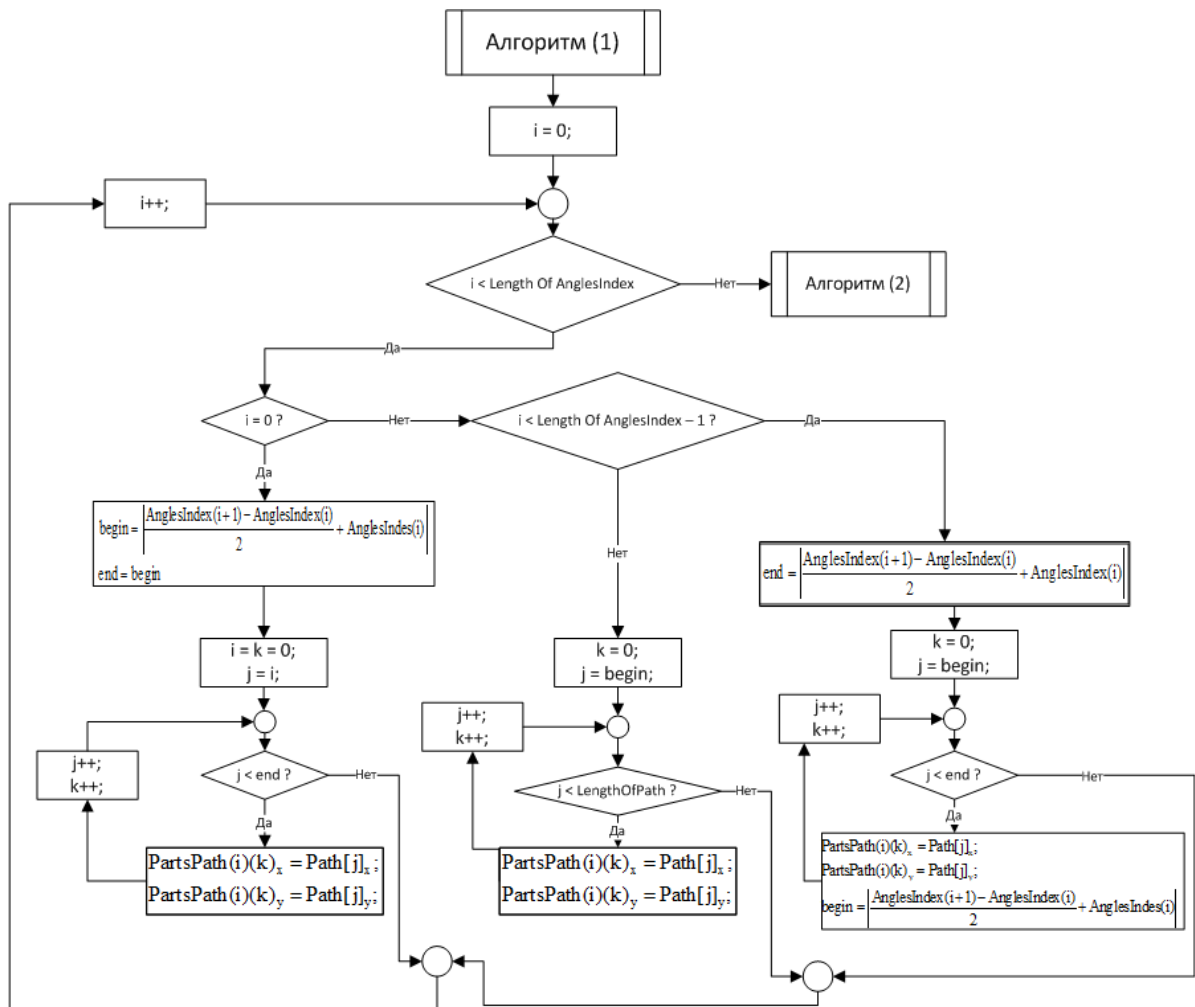


Рисунок 4.12 – Алгоритм поиска прямолинейных участков траектории

На рисунке 4.13 показана структурная схема алгоритма процесса (подпрограммы) процесса поиска прямолинейных участков пути в рабочей плоскости промежуточного положения диагональным методом. Порядок вызова этой подпрограммы *Алгоритм (2)* \Rightarrow *Проведения расчетов радиусов сопрягающих окружностей для сглаживания каждого участка пути (PartsPath)*. Целью работы Алгоритма (2) являются:

- 1) Проверка количества участков пути, которые необходимо сопрячь между собой.

2) Если количество участков (PartPath) больше начального значения, заданного в программе, переход к процессу, описанному Алгоритмом (3). Иначе осуществляется переход к Алгоритму (6).

Данный алгоритм производит проверку 4-х типов угла в рабочей плоскости диагональным методом. По приведенным ниже формулам (4.3) определяется тип угла.

$$A = \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y < Path(i+1)_y \end{cases}$$

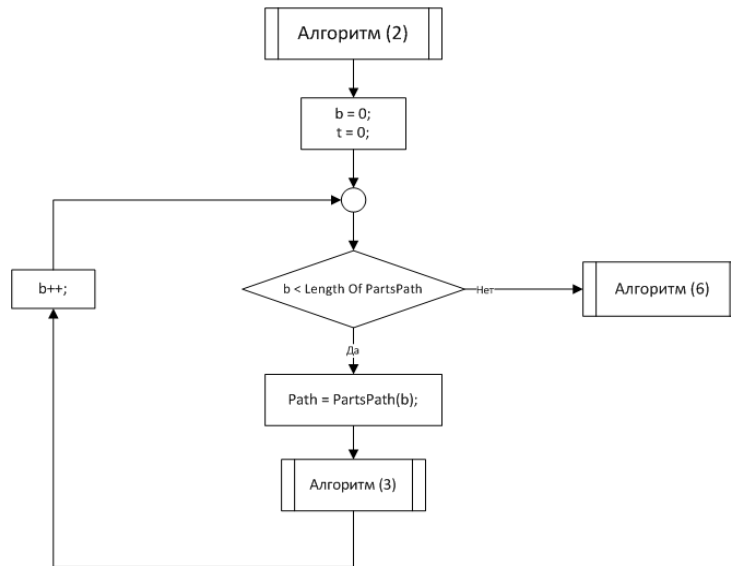


Рисунок 4.13 – Алгоритм процесса поиска прямолинейных участков пути в рабочей плоскости промежуточного положения

$$B = \begin{cases} Path(i)_x < Path(i-1)_x \text{ and } Path(i)_x = Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y < Path(i+1)_y \end{cases}$$

$$C = \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases}$$

$$D = \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y > Path(i+1)_y \end{cases} \quad (4.3)$$

Далее в соответствии с выявленным типом угла по нижеприведенным формулам определяются координаты начальной и конечной точки угла соответствующего типа.

$$Checking1 = \begin{cases} (x \geq StartPoint_x \text{ and } x \leq EndPoint_x) \\ AND \\ (y \leq StartPoint_y \text{ and } y \geq EndPoint_y) \end{cases} \quad : \text{ для типа угла (A)}$$

$$Checking2 = \begin{cases} (x \geq StartPoint_x \text{ and } x \leq EndPoint_x) \\ AND \\ (y \geq StartPoint_y \text{ and } y \leq EndPoint_y) \end{cases} \quad : \text{ для типа угла (B)}$$

$$Checking3 = \begin{cases} (x \leq StartPoint_x \text{ and } x \geq EndPoint_x) \\ AND \\ (y \geq StartPoint_y \text{ and } y \leq EndPoint_y) \end{cases} \quad : \text{ для типа угла (C) и (D)} \quad (4.4)$$

На рисунке 4.14 показана структурная схема алгоритма процесса (подпрограммы) процесса формирования сопрягающей дуги в рабочей плоскости промежуточного положения диагональным методом. Порядок вызова этой подпрограммы **Алгоритм(3) → Нахождение типа углов (*AngleTypes*) и формирование сопрягающей дуги**. Целью работы Алгоритма (3) являются:

1) Определение тип угла.

2) После выявления одного из 4 типов углов, определяются необходимые данные (начальная, конечная и центральная точки) для формирования окружности сопряжения, вписанной в этот угол. В данном алгоритме используется метод построения окружности по трем точкам.

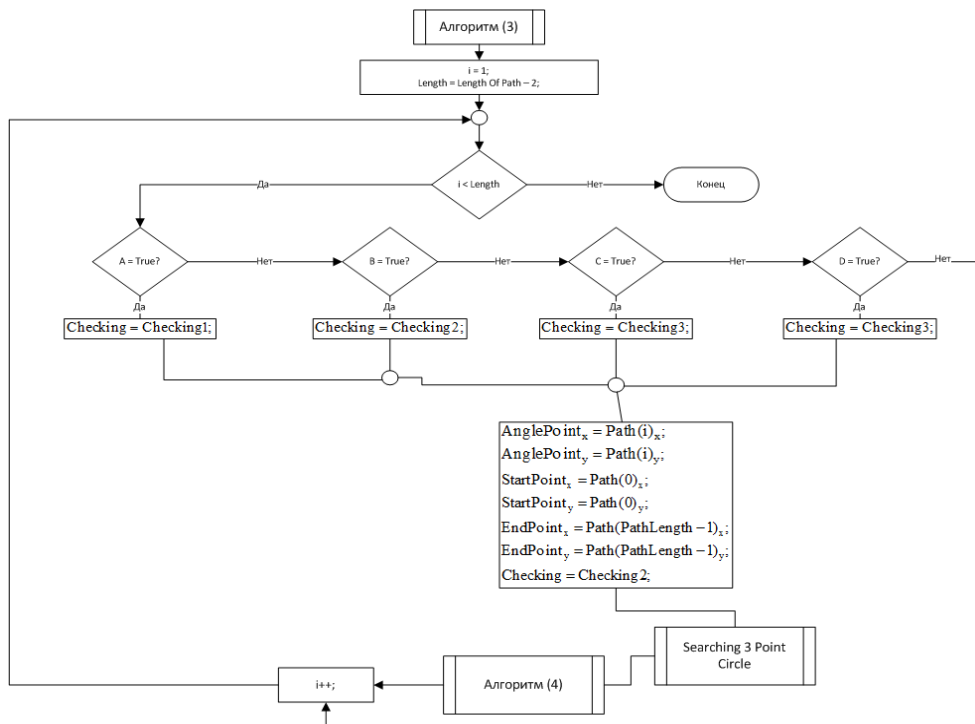


Рисунок 4.14 – Определение одного из 4-х типов углов

На рисунке 4.15 показана структурная схема алгоритма процесса (подпрограммы) процесса формирования сопрягающей дуги в рабочей плоскости промежуточного положения диагональным методом. Порядок вызова этой подпрограммы **Алгоритм (4) → формирование дуги и получение плавной кривой (NewPath)** для каждого угла. Алгоритм (4) используется для построения дуги окружности по известным формулам. Однако для построения плавной траектории будет использован лишь необходимый для сглаживания пути участок этой дуги (NewPath).

На рисунке 4.16 показана структурная схема алгоритма процесса (подпрограммы) процесса выбора отрезков, смежным по отношению к ранее определенным углам в рабочей плоскости промежуточного положения. Порядок вызова этой подпрограммы **Algorithm (5) → Получение нормального гладкого пути (NewPath) для каждого угла.** Данный алгоритм находит прямой участок пути возле угла в текущем отрезке пути в рабочей плоскости.

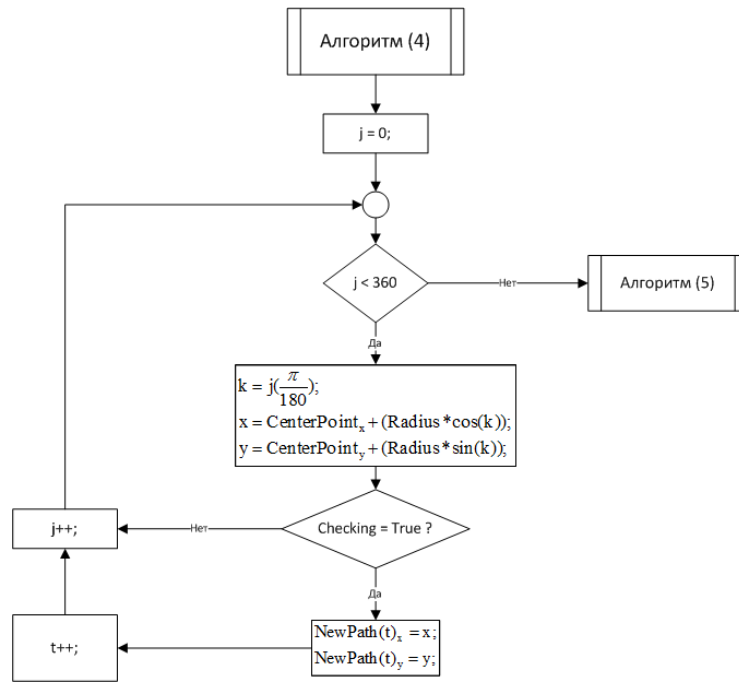


Рисунок 4.15 – Алгоритм формирования дуги окружности и получения плавной кривой

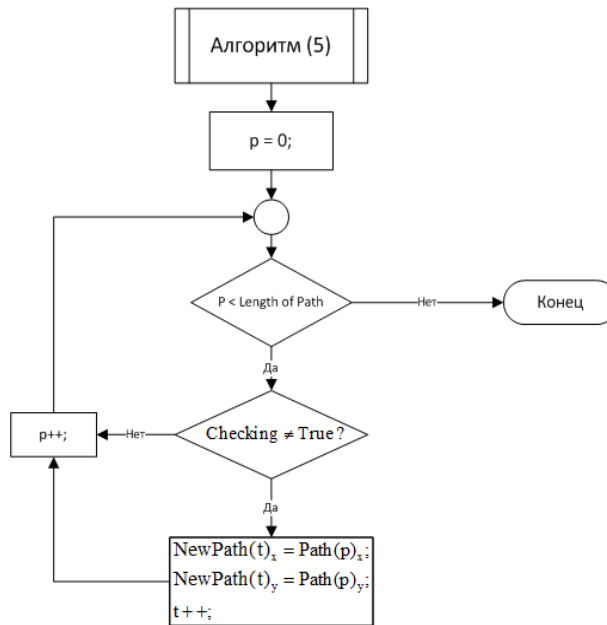


Рисунок 4.16 – Алгоритм процесса выбора отрезков, смежным по отношению к ранее определенным углам

На рисунке 4.17 показана структурная схема алгоритма процесса (подпрограммы) процесса выбора отрезков, смежным по отношению к ранее определен-

ным углам в рабочей плоскости промежуточного положения. Порядок вызова этой подпрограммы *Algorithm (6) → Формирование комбинации дуги и отрезка для каждого угла для получения плавной траектории*. Алгоритм (6) используется, чтобы сформировать новую гладкую траекторию в рабочей плоскости.

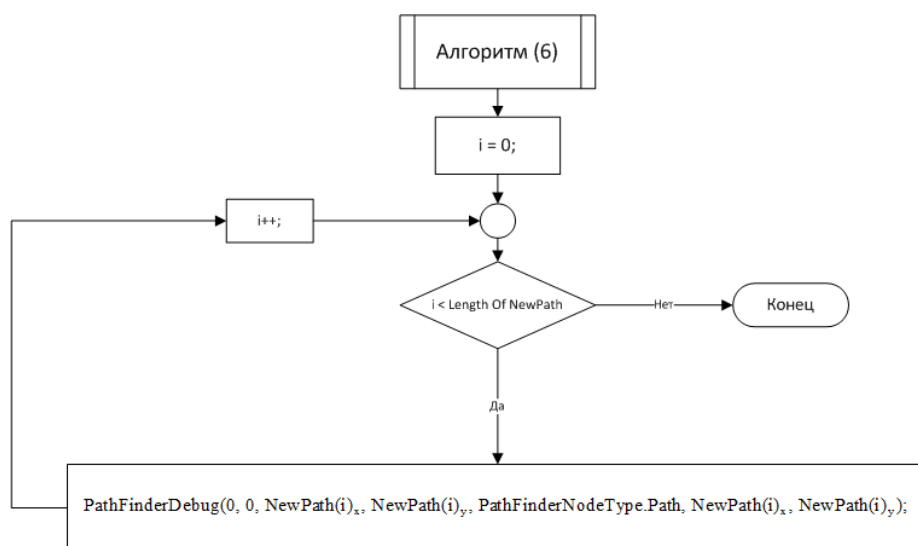


Рисунок 4.17 – Формирование комбинации дуги и отрезка для каждого угла

На рисунке 4.18 показана структурная схема алгоритма процесса (подпрограммы) сглаживания траектории в пространстве (3D-случай). Порядок вызова этой подпрограммы *Главная программа → Нахождение углов (AnglesIndex)*. С помощью этой подпрограммы возможно осуществить:

1) Проверку длины траектории, полученной с помощью улучшенного алгоритма A* в 3D пространстве. Если длина траектории больше начального значения, заданного главной программой, переходим к процессу, описанному алгоритмом (1).

2) Проверка углов для последующего сглаживания траектории.

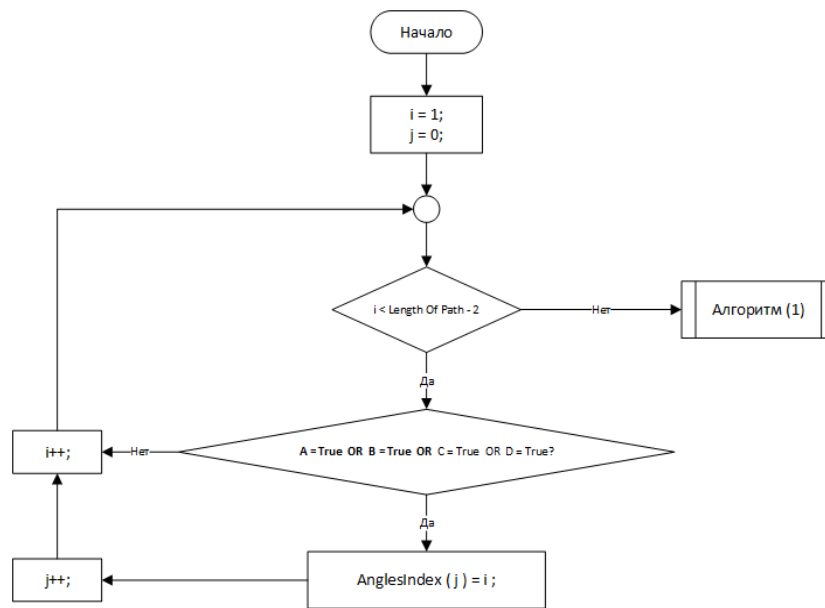


Рисунок 4.18 – Алгоритм подпрограммы сглаживания пути в 3D пространстве

На рисунке 4.19 показана структурная схема алгоритма процесса (подпрограммы) сглаживания траектории в пространстве (3D-случай). Порядок вызова этой подпрограммы *Алгоритм (1) → Поиск участков пути (PartsPath)*. Алгоритм (1) позволяет:

- 1) Обнаружить прямые участки пути в 3D - пространстве.
- 2) Установить начальную и конечную точку прямого участка пути до и после каждого угла.

На рисунке 4.20 показана структурная схема алгоритма процесса (подпрограммы) сопряжения отрезков в пространстве (3D-случай) дугами окружностей. Порядок вызова этой подпрограммы *Алгоритм (2) → Расчет для сглаживания каждого участка пути (PartsPath)*. Назначение Алгоритма (2):

- 1) Проверка количества прямолинейных участков трассы.
- 2) Если количество прямолинейных участков (PartPath) больше начального значения, заданного в программе, осуществляется переход к процессу, описанному Алгоритмом (2). Иначе происходит переход к алгоритму (8).

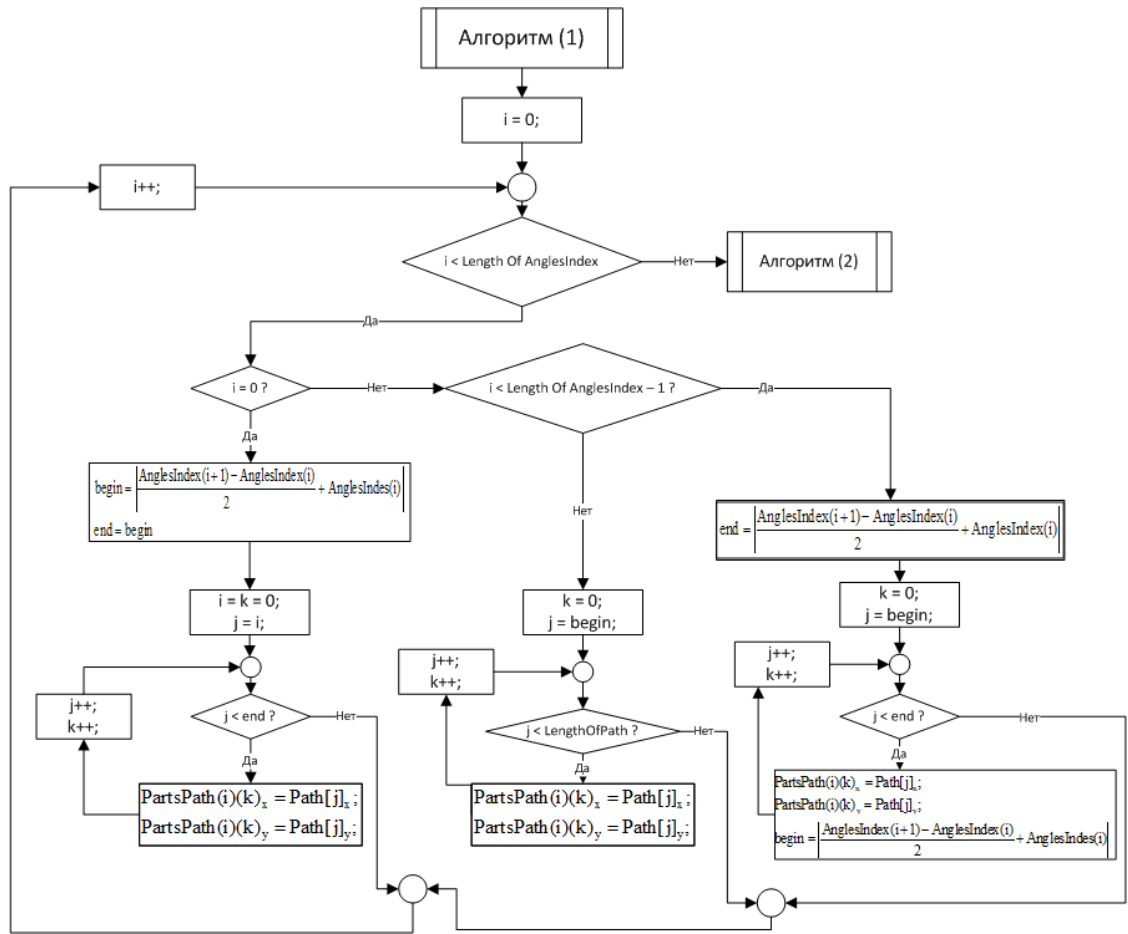


Рисунок 4.19 – Алгоритмы поиска участков пути в 3D пространстве

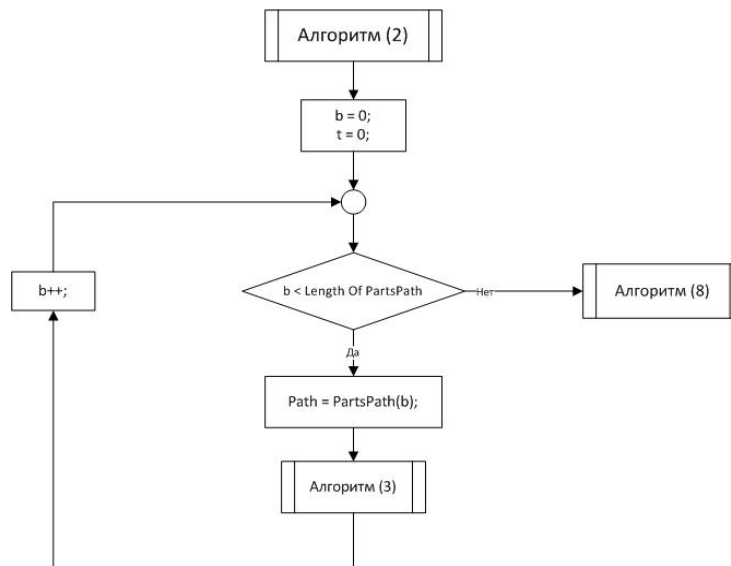


Рисунок 4.20 – Алгоритм проверки количества участков пути в 3D

Работа этого алгоритма предусматривает проверку типа угла в некоторой рабочей плоскости в пространстве, что в дальнейшем позволит по результатам проверки типа угла сформировать гладкую траекторию в 3D – пространстве. Тип угла определяется по приведенным ниже формулам тип угла (4.5).

$$\begin{aligned}
 A &= \begin{cases} Path(i)_x = Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases} \\
 B &= \begin{cases} Path(i)_x = Path(i-1)_x \text{ and } Path(i)_x > Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases} \\
 C &= \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x = Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y > Path(i+1)_y \end{cases} \\
 D &= \begin{cases} Path(i)_x < Path(i-1)_x \text{ and } Path(i)_x = Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y > Path(i+1)_y \end{cases} \\
 E &= \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y < Path(i+1)_y \end{cases} \\
 F &= \begin{cases} Path(i)_x < Path(i-1)_x \text{ and } Path(i)_x = Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y < Path(i+1)_y \end{cases} \\
 G &= \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y < Path(i-1)_y \text{ and } Path(i)_y = Path(i+1)_y \end{cases} \\
 H &= \begin{cases} Path(i)_x > Path(i-1)_x \text{ and } Path(i)_x < Path(i+1)_x \\ AND \\ Path(i)_y = Path(i-1)_y \text{ and } Path(i)_y > Path(i+1)_y \end{cases} \quad (4.5)
 \end{aligned}$$

Для того, чтобы осуществить проверку типа угла необходимо предварительно сформировать его смежные отрезки в пространстве по приведенным ниже

формулам (4.6), которые определяют координаты начальной и конечной точки для 8 типов углов:

$$\begin{aligned}
 \text{Checking1} &= \begin{cases} (x \geq \text{StartPoint}_x \text{ and } x \leq \text{EndPoint}_x) \\ \text{AND} \\ (y \leq \text{StartPoint}_y \text{ and } y \geq \text{EndPoint}_y) \end{cases} \\
 \text{Checking2} &= \begin{cases} (x \geq \text{StartPoint}_x \text{ and } x \leq \text{EndPoint}_x) \\ \text{AND} \\ (y \geq \text{StartPoint}_y \text{ and } y \leq \text{EndPoint}_y) \end{cases} \\
 \text{Checking3} &= \begin{cases} (x \leq \text{StartPoint}_x \text{ and } x \geq \text{EndPoint}_x) \\ \text{AND} \\ (y \geq \text{StartPoint}_y \text{ and } y \leq \text{EndPoint}_y) \end{cases} \quad (4.6)
 \end{aligned}$$

На рисунке 4.21 показана структурная схема алгоритма процесса (подпрограммы) определения типа угла и параметров сопрягающей дуги. Порядок вызова этой подпрограммы **Алгоритм(3) → Нахождение типа углов (AngleTypes) и формирование сопрягающей дуги**. Алгоритм (3) предназначен для того, чтобы:

1) Определить тип угла.

2) Если текущий угол принадлежит одному из 8 типов, записываются необходимые данные (а именно, начальная, конечная и центральная точки) для дуги, вписанной в текущий угол.

На рисунке 4.22 показана структурная схема алгоритма процесса (подпрограммы) определения радиуса для сопрягающей дуги каждого угла. Порядок вызова этой подпрограммы **Алгоритм (4) → Нахождение радиуса для сопрягающей дуги каждого угла**. С помощью Алгоритма (4) находится радиус и центральная точка окружности, вписанной в текущий угол, для его последующего скругления траектории дугой этой окружности.

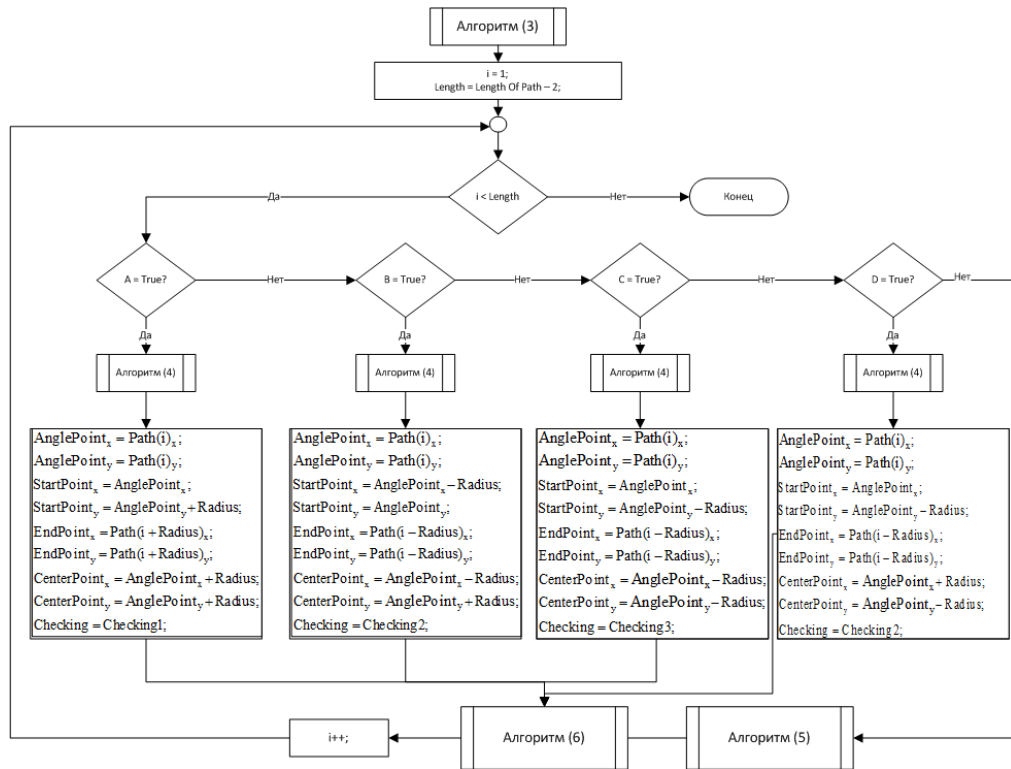


Рисунок 4.21 – Алгоритм нахождения типа углов (AngleTypes) и формирование сопрягающей дуги

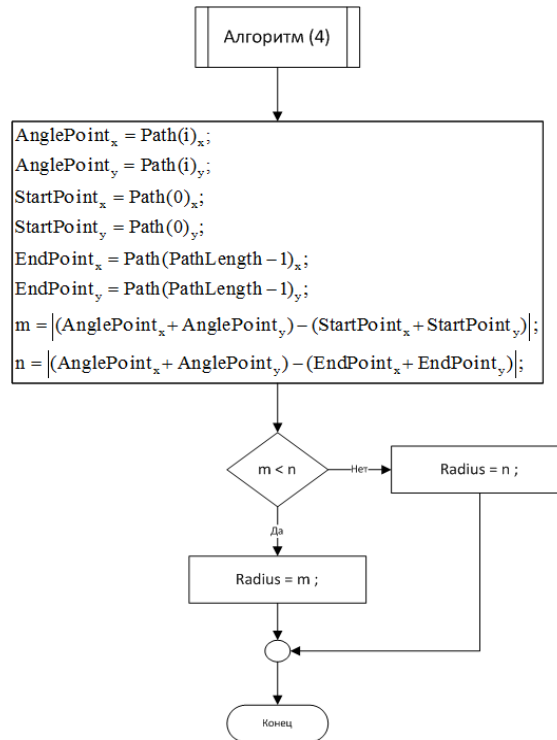


Рисунок 4.22 — Алгоритм нахождения радиуса и центральной точки окружности

На рисунке 4.23 показана структурная схема алгоритма процесса (подпрограммы) определения радиуса для сопрягающей дуги каждого угла. Порядок вызова этой подпрограммы *Algorithm (5) → Нахождение типа углов (AngleTypes)*. Алгоритм (3) предназначен для того, чтобы:

1) Определить тип угла.

2) Если текущий угол принадлежит одному из 8 типов, определяются и записываются необходимые данные (а именно, начальная, конечная и центральная точки) для формирования окружности, вписанной в текущий угол. Как и в предыдущих подпрограммах, используется метод построения окружности по трем точкам.

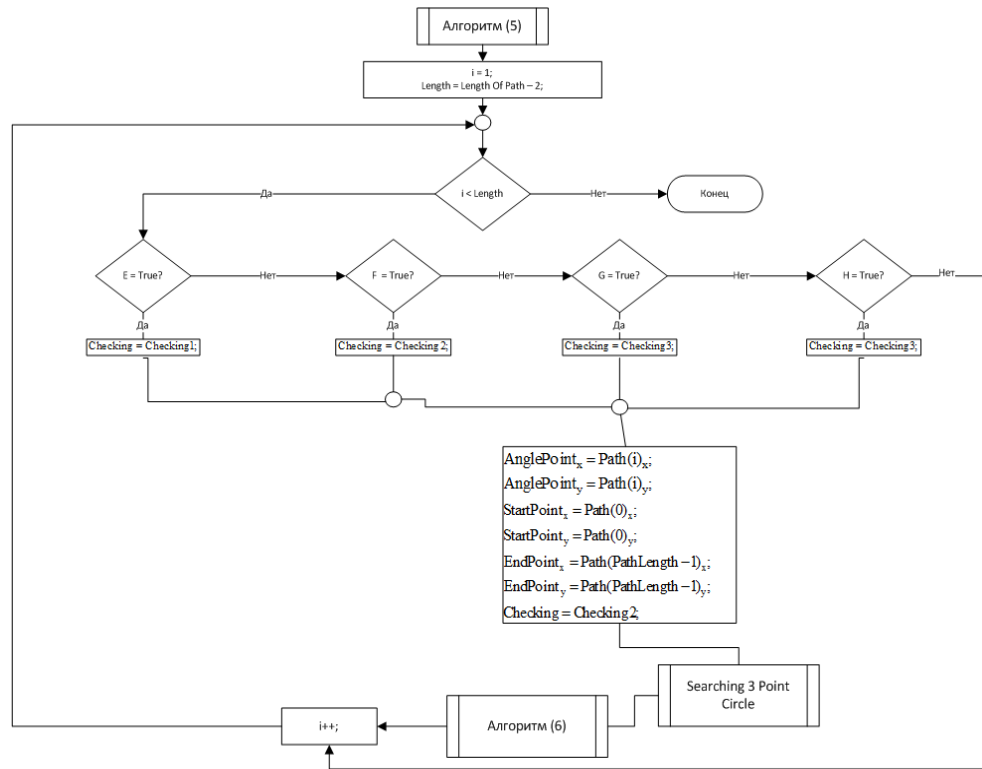


Рисунок 4.23 – Определение типов углов и параметров сопрягающей окружности

На рисунке 4.24 показана структурная схема алгоритма процесса (подпрограммы) определения радиуса для сопрягающей дуги каждого угла и сопряжения этой дугой прямолинейных отрезков угла. Порядок вызова этой подпрограммы

Algorithm (6) → Формирование окружности и получение плавной сопряжения (NewPath) для каждого угла. Алгоритм (6) используется для построения сопрягающей окружности по известным формулам аналитической геометрии. При этом используется для сглаживания лишь небольшой, необходимый для сглаживания отрезка участок сформированной этим алгоритмом окружности (NewPath).

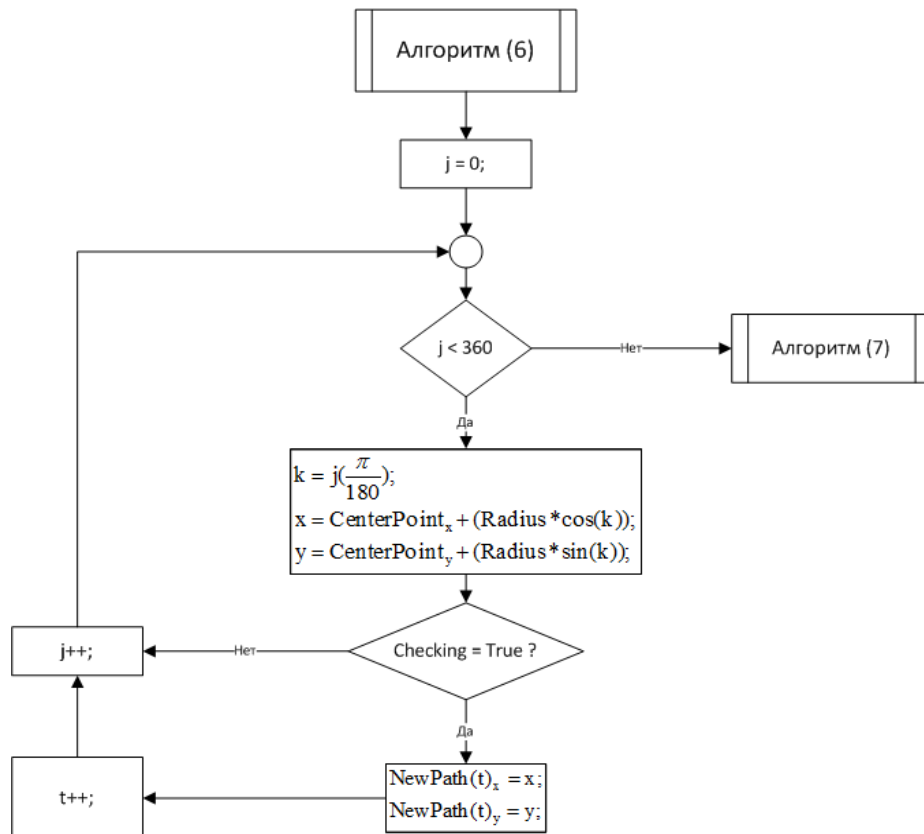


Рисунок 4.24 — Алгоритм формирования окружности и ее использования для сопряжения смежных отрезков угла

На рисунке 4.25 показана структурная схема алгоритма процесса (подпрограммы) определения выявления прямых отрезков в пространстве. Порядок вызова этой подпрограммы **Algorithm (7) → Выявление смежных отрезков в пространстве (NewPath) для каждого угла.** Данный алгоритм находит прямой участок пути возле угла в текущем отрезке манхеновой траектории в 3D – пространстве после предварительной трассировки алгоритмом А*.

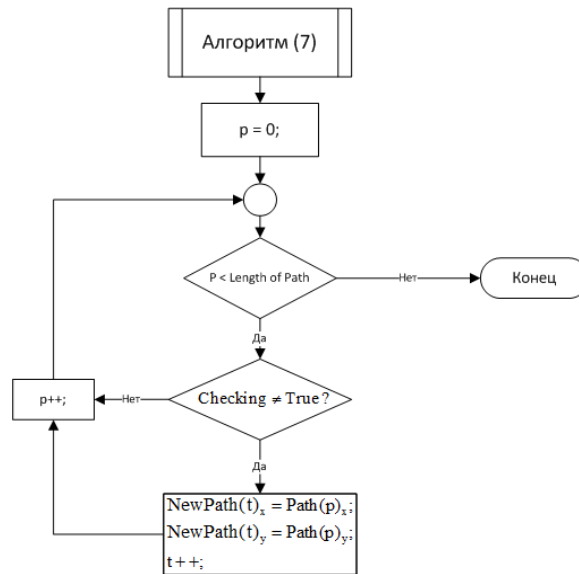


Рисунок 4.25 – Алгоритм поиска смежных отрезков в пространстве для каждого угла

На рисунке 4.26 показана структурная схема алгоритма процесса (подпрограммы) окончательной операции построения плавной траектории трассы в пространстве. Порядок вызова этой подпрограммы *Algorithm (8) → Формирование дуги и отрезка для каждого угла для получения плавной траектории.* Алгоритм (8) используется что окончательного формирования параметров новый сглаженной траектории в 3D пространстве.

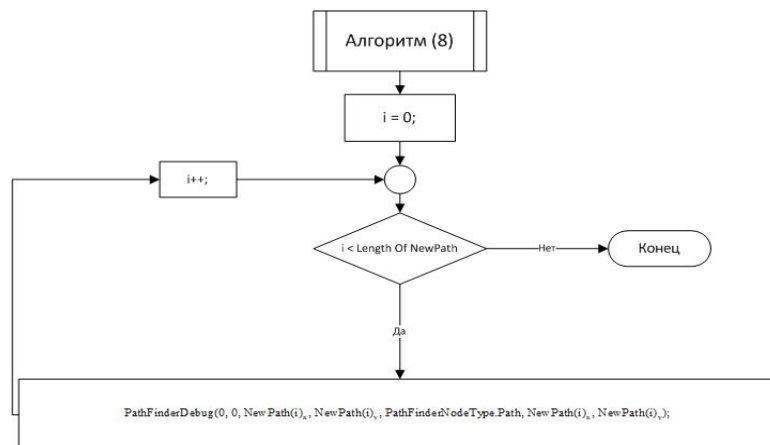


Рисунок 4.26 – Алгоритм окончательного формирования плавной траектории в пространстве

4.2 Особенности программной реализации предложенной геометрической модели телесной трассировки

Как уже отмечалось, в основу реализации предложенного метода телесной трассировки положен алгоритм A*. Текущая классическая A* модель может найти путь между двумя неподвижными объектами с нормальной скорости. Но если мы разместим много объектов в области поиска при большом количестве рецепторов в рецепторной матрице нам для практического использования будет необходим очень быстрый алгоритм. Усложнение задач, поставленных перед этим алгоритмом (построение плавной трассы и др.) требуют не только его существенной модификации по существу, но и существенной модификации информационной структуры алгоритма. При реализации оригинального алгоритма A* на языке программирования C# можно заметить, что наибольшее время работы программы занимает нахождение пути на участках большой длины или в случае, когда требуемую траекторию и вовсе невозможно найти. Выход из этой ситуации мы видим в изменении структуры данных классического алгоритма A*.

Оптимизация структуры данных программы проведена следующим образом. Известно, что в структуре данных программ алгоритма типа A* имеются **открытые** и **закрытые списки**, которые «убивают» алгоритм (в алгоритме A*, в основном используют такие два таблицы.). Главной проблемой увеличения скорости работы программы является то, что когда размер рецепторов большой (1000 × 1000 и более), число открытых и закрытых рецепторов в этих списках слишком велико, и при любом используемом методе поиск в этих списках занимает очень много времени. Изначально была рассмотрена возможность использовать *классы*, позволяющие хранить информацию об рецепторах (их кодах). Но этот путь приведет к необходимости постоянного слежения за «**сбором мусора**», так как придется следить за распределением памяти для активных рецепторов и очистке па-

мяти об уже использованных алгоритмом рецепторах. Вместо классов использовались нами при программировании использовались **структуры**, повторный вызов которых в коде приводил к улучшениям.

В процессе работы нам необходимо находить путь и содержание кода рецепторов **открытого списка - OpenList()** для того, чтобы найти рецептор поближе к конечной точке. Рецепторы будем хранить при помощи рецепторной матрицы. Зная положение (X/Y/Z), можно сразу же (с помощью **открытого списка - OpenList()**) находить нужный рецептор.

Такой подход также позволяет избежать использования **закрытых списков**, так как закрытые рецепторы (с кодом "1") можно помечать сразу же на рецепторной матрице. Поскольку каждый анализируемый рецептор связан с родительским рецептором, то поэтому **закрытый список** для реализации нашего алгоритма не нужен. Однако, совсем избавиться от открытого списка не получится, так как необходимо найти рецепторы с наименьшим весом (F), а **расчетной список** не может с этим справиться.

В этот раз воспользуемся **открытым списком** (очередью по приоритету) только для того, чтобы заносить в него и находить в нем рецептор с наименьшим весом. Для этого лучше всего подходит очередь по приоритету, так как поиск в ней не требует линейного доступа (linear access). Единственный недостаток такого подхода заключается в огромном потреблении памяти, так как приходится хранить **расчетной список**, потому что каждая ячейка хранит рецептор размером 32 байта. Все равно в принципе для рецепторной матрицы (1000x1000) необходимо 32 Мб оперативной памяти для классического алгоритма A^* . Но на **расчетном списке** есть доступ по координатам (X/Y), поэтому нет необходимости хранить эти значения в рецептором. Это уменьшает $8 \text{ байт} * 1\,000\,000$, то есть сохраняет 8 Мб оперативной памяти.

Каждый рецептор соединен с родительским узлом и является переменной типа **int (32 bits)**. Так как размер рецептора в ОЗУ не может быть слишком боль-

шой, заменим его тип данных на **ushort (16 bits)**. Это сэкономит 2 байта для каждого рецептора, а, значит, 2 Мб в сумме. Можно заметить, что **heuristic (H)** вычисляется для каждого рецептора динамически и больше не используется в алгоритме, поэтому её тоже можно убрать из структуры. **Heuristic** имеет тип **int (4 bytes)**. Таким образом, освобождается еще 4 Мб. В итоге минимальный рецептор занимает 13 байт, но из-за *единого формата данных* на него приходится 16 байт во время работы программы (рисунок 4.27). Необходимо описать структуру программы для использования единого формата данных для каждого байта.

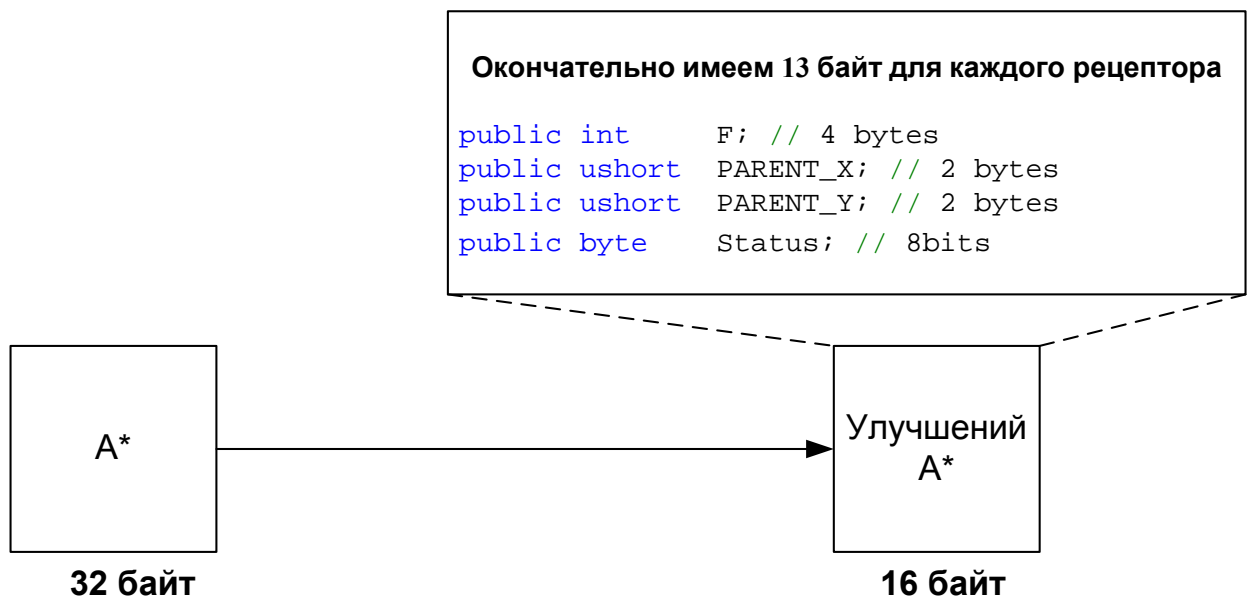


Рисунок 4.27 – Определение формата данных при работе с рецепторной матрицей

Окончательно имеем 13 байт для каждого рецептора. Поэтому во время запуска примера можно заметить, что **PathFinder_APS** занимает 13 Мб дополнительной памяти. Потребное количество оперативной памяти приведено в таблицах 4.1 и 4.2.

Таблица 4.1 – Необходимое количество оперативной памяти для 2D матрицы

№	Количество рецепторов	Размер оперативной памяти (при расчете рецепторов в 2D матрице)
1	1024x1024	13 Мб
2	512x512	3.2 Мб
3	256x256	832 Кб

Таблица 4.2 – Необходимое количество оперативной памяти для 3D матрицы

№	Количество рецепторов	Размер оперативной памяти (при расчете рецепторов в 3D матрице)
1	1024x1024	953.67 Мб
2	512x512	128 Мб
3	256x256	16 Мб

Стоит признать, что улучшенный нами алгоритм А* потребовал для своей реализации значительного времени из-за сложности его отладки. Однако из-за оптимизации его информационной структуры его быстродействие намного увеличилось. Теперь наш улучшенный алгоритм А* с новой структурой данных работает как минимум в 300 раз быстрее на простых рецепторах и в более, чем 1200 раз быстрее на сложных рецепторах.

Другой подход к оптимизация алгоритма и реализующей его программы состоит в том, что если текущий открытый рецептор (в качестве которого выступает вершина поиска) уже находится в **открытом списке**, а его суммарный вес меньше, чем у хранимого в списке, тогда рецептор должен быть заменен. Вместо этого старый открытый рецептор остается в открытом списке, а к нему добавляется еще один. Старый рецептор обладает большим весом, поэтому он будет обработан позже, но когда обработаются следующие рецепторы, он будет закрыт. Этот про-

цесс происходит быстрее, чем удаление старого открытого узла из открытого списка.

Следующее улучшение в том, между вызовами нахождения пути необходимо очистить расчетную сетку. Рецепторная матрица хранит объект типа **PathFinder_Node**, а каждый объект имеет поле **Status**; это поле показывает открыт или закрыт рецептор, или и тот и другой случай. Понятно, что закрытый рецептор является для нас зоной запрета и движение в его направлении не производится.

Значение статуса могут быть **0** = не обработан, **1** = открыт, **2** = закрыт. Использование параметра “Статус” позволяет по-новому перейти к работе с рецепторными моделями, перейдя от однозначной логики (“0” и “1”) к многозначной (“0”, “1” и “2”).

Обнуление рецепторной матрицы занимает около 30 миллисекунд для рецепторов размером 1024x1024 и происходит между вызовами PathFinder_APS.

Чтобы это оптимизировать, вместо обнуления рецепторной матрицы между вызовами изменим значения поля статус у рецептора. Тогда, значение увеличится на 2.

Тогда при первом поиске пути имеем:

- 1 = Открытый
- 2 = Закрытый
- x = Нет проверил

При втором поиске пути:

- 3 = Открытый
- 4 = Закрытый
- x = Нет проверил

И так далее.

Таким образом, между вызовами PathFinder_APS нет необходимости обнулять рецепторную матрицу. Возможно использовать различные величины для от-

крытых и закрытых рецепторов. Остальные значения буду означать отсутствие исследования.

Еще одна небольшая оптимизация заключается в переходе всех **локальных переменных (local variables)** в переменные-члены (**member variables**). Это позволит создавать все переменные сразу в **куче (heap)**, а не создавать/ уничтожать их в стеке (**stack**) для каждого вызова нахождения пути.

В ходе оптимизации удалось избавиться от распределения памяти и последовательного поиска для фиксированного распределения и минимального поиска.

Подведем итог внесенных нами улучшений в алгоритм трассировки. Это:

1. **Генерация переключателя выбора направления поиска.** Если переключатель настроен на поиск пути по манхэттенскому направлению, то поиск будет вестись в 4-х направлениях в режиме 2D и по 8 направлениям в режиме 3D. Если же переключатель настроен на поиск диагонального направления, то в режиме 2D поиск будет вестись по 8 направлениям в режиме 3D – по 26 направлениям.

2. **Штраф за смену направления.** Если флаг этой опции установлен, то каждый раз при смене направления стоимость пути уменьшается и в итоге спроектированная трасса получается более гладкой, без большого количества смены направлений. Недостаток включения этой опции заключается в том, что нахождение такой трассы займет больше времени, так как придется проверять дополнительные узлы.

3. **Пересмотр закрытых узлов.** Предложенный алгоритм разрешает пересматривать уже пройденные узлы, если их стоимость меньше значения стоимости предыдущего узла. Это также позволяет улучшить путь, сделав его более гладким, но для этого потребуется больше расчетного времени.

4. **Эвристика.** Измененная нами эвристика влияет на оценку расстояния от текущей позиции до целевого узла. Введенная нами эвристика позволяет рассчитать, сколько понадобится времени, чтобы достичь цели. Чем выше эта оценка, тем короче найденный путь.

5. Размер рецептора. Этот параметр является настраиваемым и определяется посредством интерфейса программы. Уменьшение размера рецептора повышает точность геометрической модели, но требует значительно больших вычислительных ресурсов для проектирования одной и той же трассы. Выбор разумного компромисса между точностью геометрической модели и необходимым компьютерными ресурсами для ее реализации является предметом специального исследования.

6. Быстрое нахождение пути. Это опция алгоритма, устанавливаемая посредством интерфейса программы. Когда флаг этой опции снят, при реализации программы используется классический алгоритм A^* (на предварительном этапе проектирования отрезков или если не требуется проведение плавных трасс). В противном случае используется улучшенный алгоритм A^* . Он требует больше памяти, но и работает примерно в 300-1200 раз быстрее в зависимости от сложности компоновочного пространства.

4.3 Интерфейс программы, реализующий алгоритм трассировки

Для реализации предложенного алгоритма трассировки была написана программа Advanced Pathfinder System (APS) в Microsoft Visual Studio 2010, используя язык программирования C#. На рисунке 4.28 показан интерфейс разработанной нами программы Advanced Pathfinder System (APS), которую можно использовать на двух языках – английском и русском. С помощью этой программы можно:

1. Используя улучшенный алгоритм A^* , найти рациональную трассу между двумя точками в 2D и 3D пространствах с учетом областей запретов;
2. Осуществить сглаживание полученной на предыдущем этапе трассы либо на любой заданный радиус, либо на максимально возможный радиус с после-

дующей проверкой выполнения условия, что минимальный радиус не меньше заданного R_{min} ;

3. Обеспечивать прохождение трассы на заданном минимальном расстоянии δ от уже размещенных объектов и областей запрета.

Примеры работы этой программы показаны на следующих иллюстрациях. Как уже отмечалось, сначала разработанный нами алгоритм и реализующая его программа работает как алгоритм A^* , проводя трассу отрезками прямых (рисунок 4.29 а), а затем скругляются радиусом, максимально возможным в данном случае (рисунок 4.29 б). Если визуализация первоначальной трассы (отрезки прямых) не требуется, их можно скрыть.

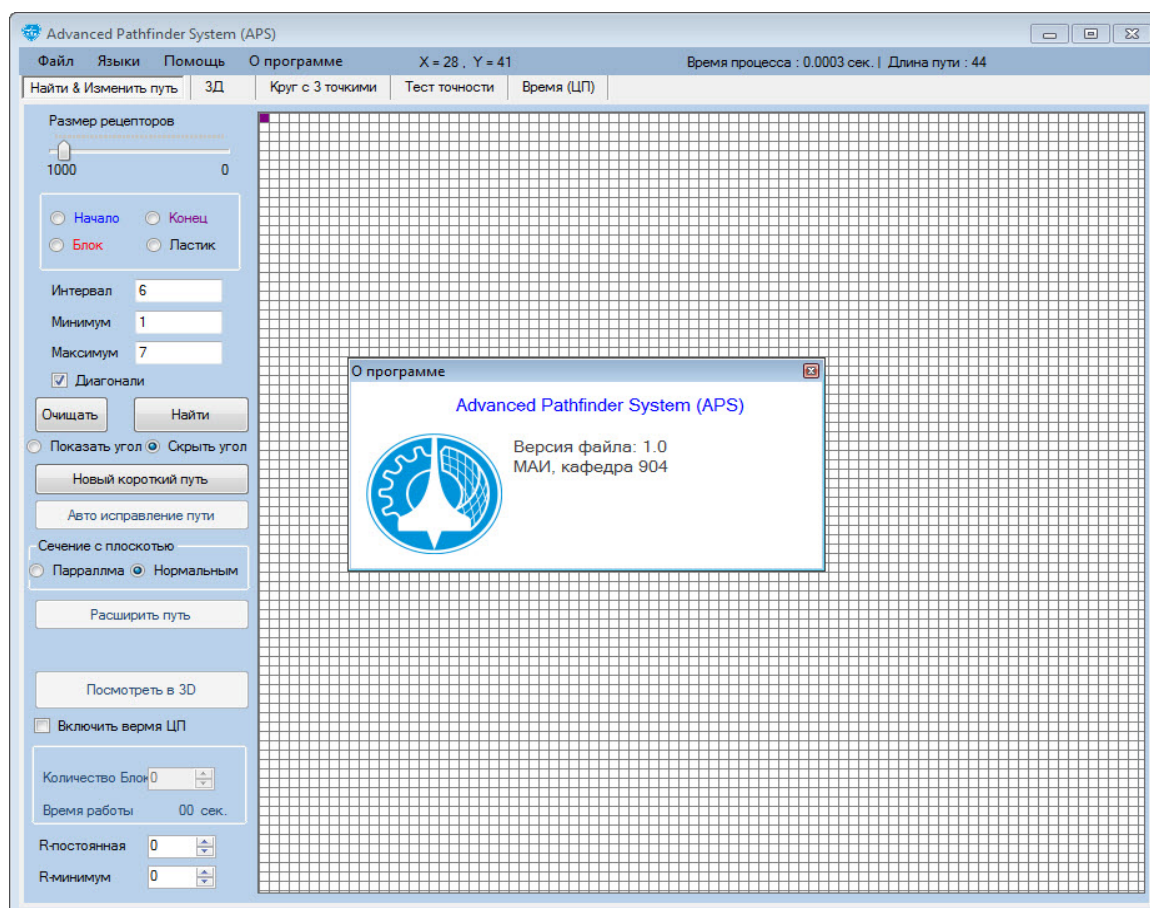
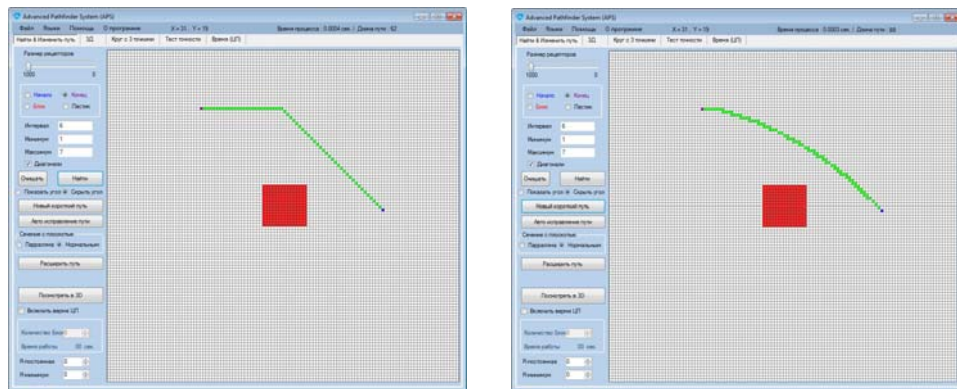


Рисунок 4.28 - Интерфейс разработанной нами программы Advanced Pathfinder System (APS)



a)

б)

Рисунок 4.29 – Этапы работы программы APS: а – предварительное представление трассы отрезками прямых; б - скругление трассы максимально возможным радиусом

На рисунке 4.30 показаны этапы работы программы APS – построение сначала манхэттенской трассы, затем ее скругление заданным радиусом $R=10$ и $R=8$ (на рисунке показан 2D случай).

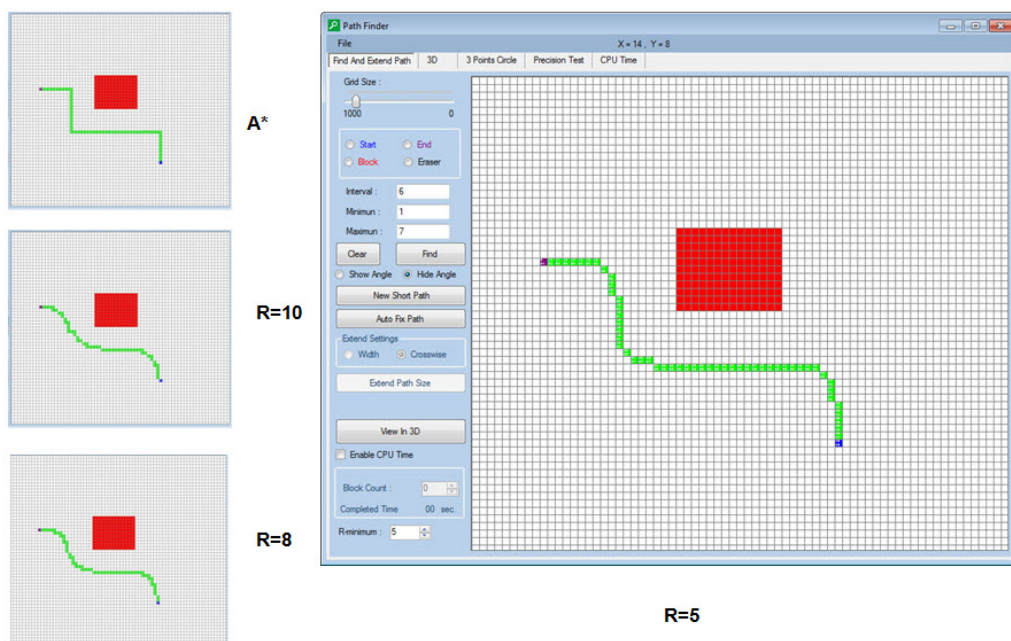


Рисунок 4.30 – Иллюстрация работы программы APS (скругление трассы разными радиусами)

На рисунке 4.31 показана работа программы по построению трассы с обеспечением заданного расстояния δ от области запрета. На этом рисунке видно, что такой режим работы алгоритма, и соответственно программы, достигается “разбуханием” областей запрета на заданную величину δ и затем использованием нашего алгоритма в его традиционном виде.

На рисунке 4.32 показано построение телесной трассы заданного размера в плоскости и пространстве, что заключается в построении главной направляющей линии канала (ГНЛ) и ее последующем “разбухании” до заданного сечения.

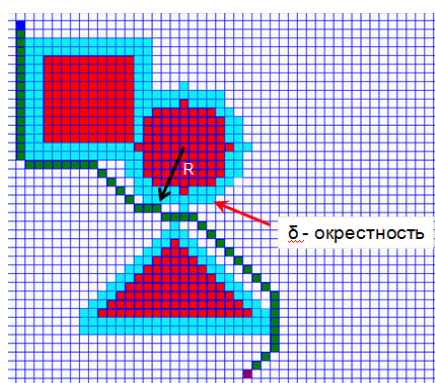
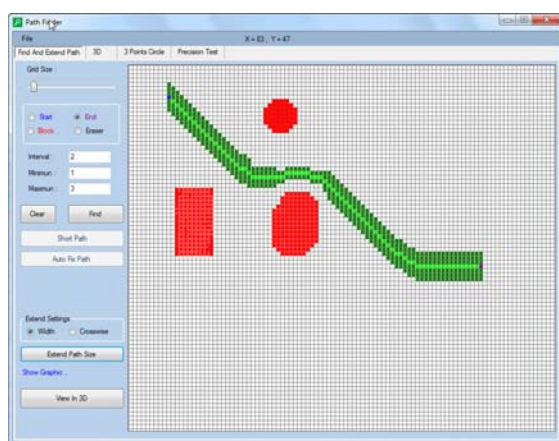
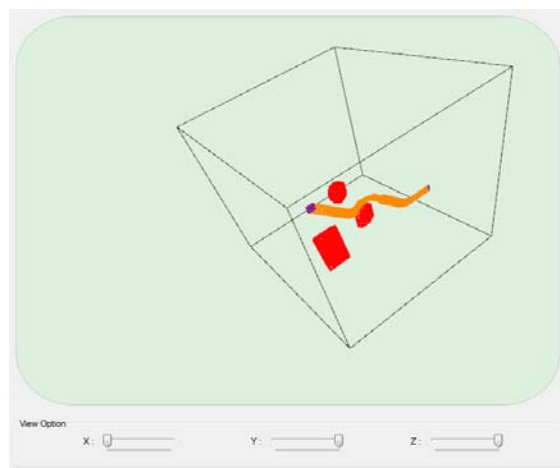


Рисунок 4.31 – Проведение трассы в заданной δ окрестности от областей запрета



a)



b)

Рисунок 4.32 – Проектирование телесной трассы: a – плоский случай; б – пространственный случай

Параметры трассы (начальная и конечная точка, радиусы кривизны, площадь сечения трассы и др.), а также варианты просмотра готового компоновочного решения настраиваются посредством интерфейса программы (рисунок 4.33). Этапы построения плавной телесной трассы показаны на рисунке 4.34. На этом рисунке показан пример поиска рациональной трассы из начальной в конечную точки в 3D пространстве при помощи двух методов – диагонального и по методу правого угла.

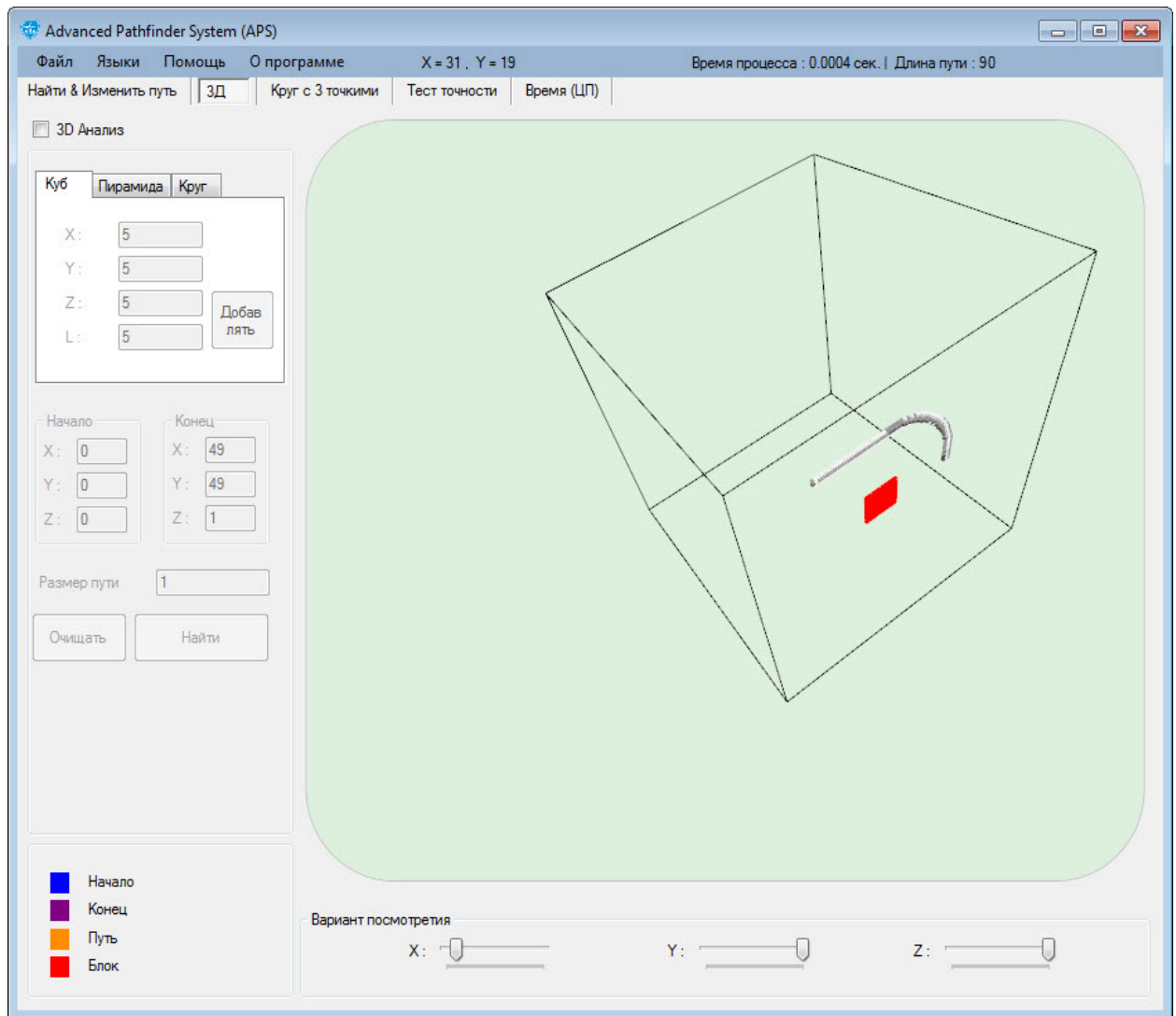


Рисунок 4.33 – Настройка параметров проектируемой телесной трассы и вариантов ее просмотра посредством интерфейса программы

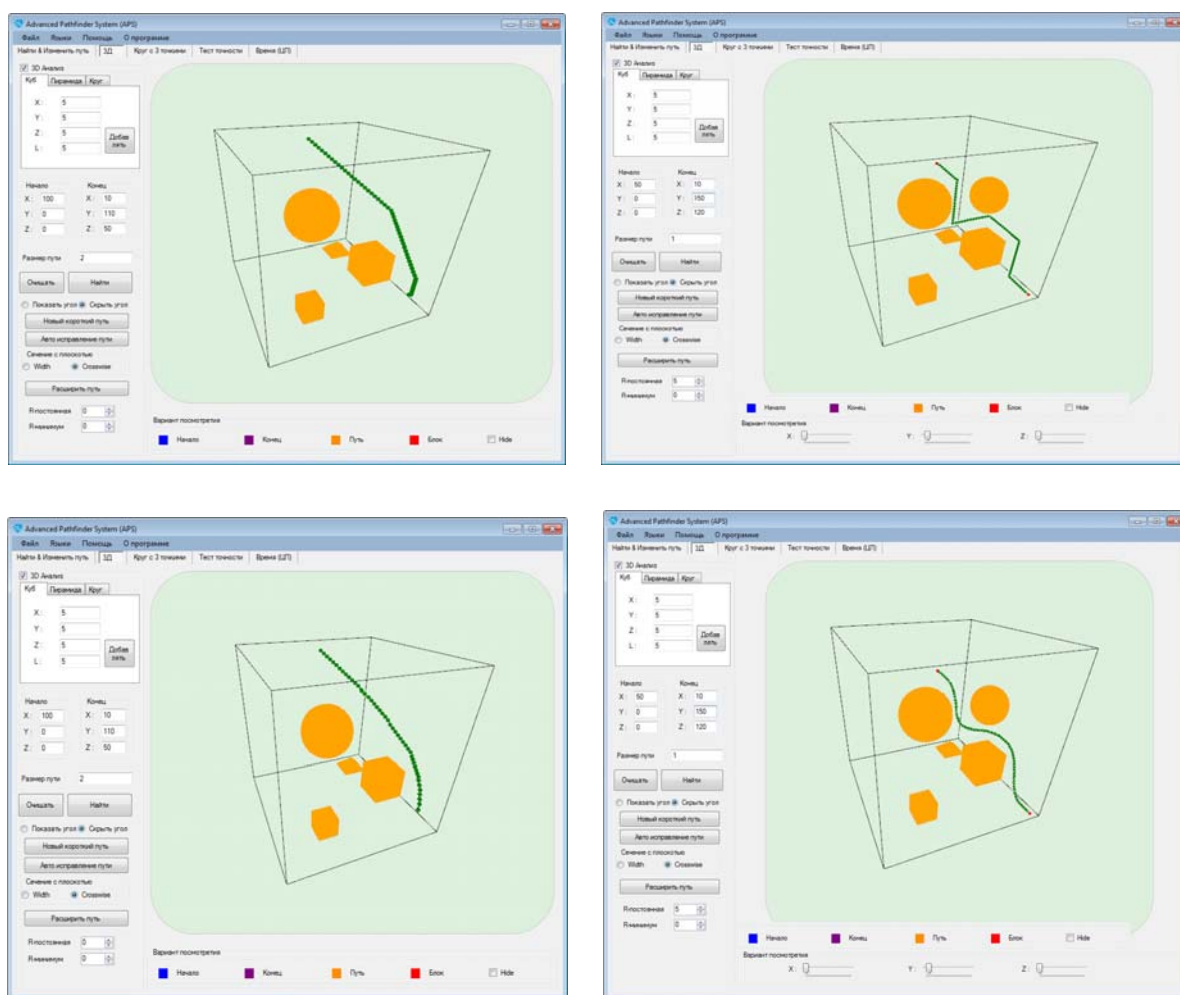
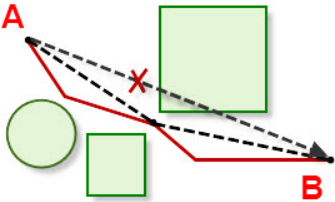
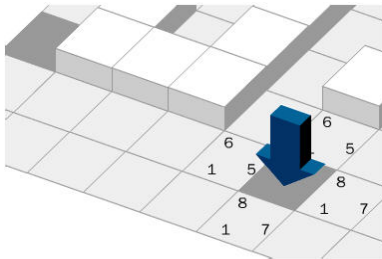


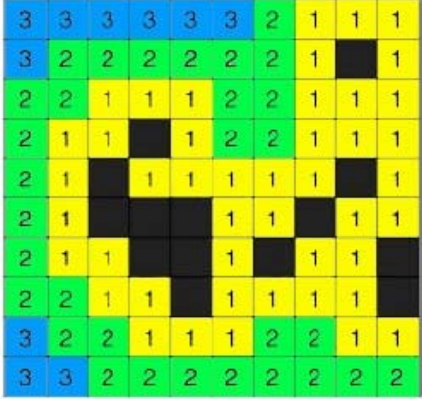
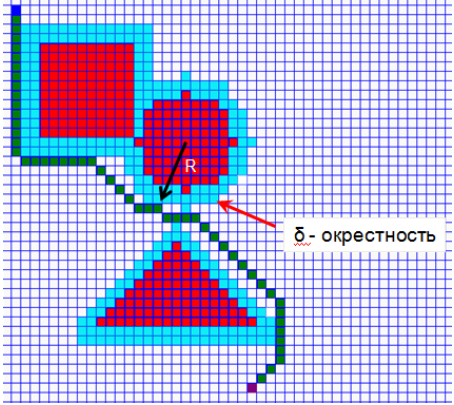
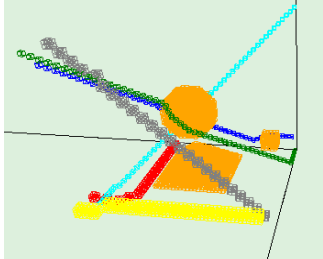
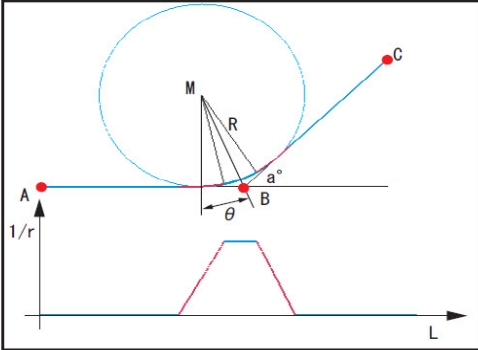
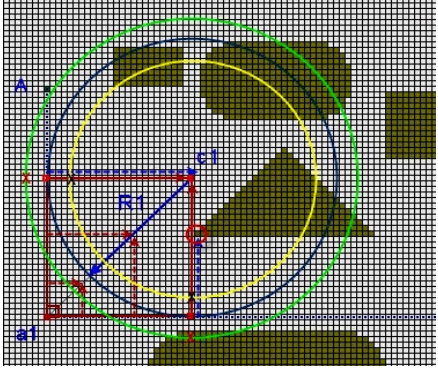
Рисунок 4.34 – Этапы построения плавной пространственной телесной трассы

4.4 Исследование и верификация алгоритма и программы телесной трассировки

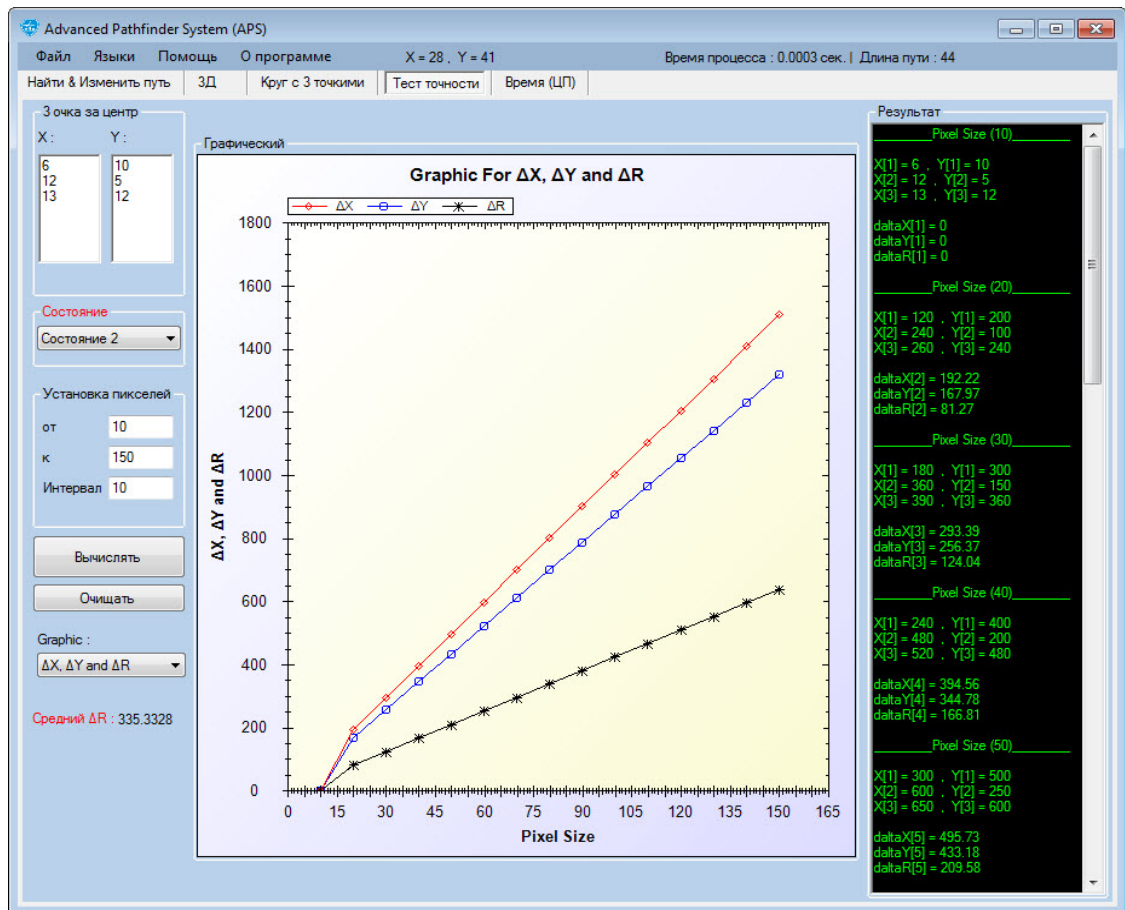
Начнем исследование разработанного алгоритма телесной трассировки сравнивая его с наиболее близким по используемым приемам рецепторным методом Масамото Канехары [226], разработанным для формирования рациональных движений робота-манипулятора. Результаты сравнения этих алгоритмов приведены в таблице 4.3.

Таблица 4.3 – Сравнение алгоритмов Канехары и разработанного алгоритма

No.	Алгоритм Масатомо Канехары	Алгоритм, используемый в данной работе
1	<p>Сокращение пути между двумя узлами с помощью метода выборки (методом A* выбираются 2 узла рецепторной модели и создается кратчайший путь. Если между 2 узлами есть препятствие, путь отбрасывается. Процедура повторяется до тех пор, пока путь A* не становится минимальным).</p> 	<p>Пусть укорачивается за счет штрафов за смену направления (Каждый раз, когда алгоритм меняет направление, вес пути уменьшается. В результате путь меняет направление небольшое число раз и выглядит естественнее).</p> 
2	<p>На рецепторной матрице зона безопасности движений робота обозначена разными цветами, которые обозначают расстояние до препятствия. Чем темнее цвет, тем ближе запрет, чем светлее – тем дальше.</p>	<p>Задаваемая δ окрестность для трубопровода или проводов достигается путем добавления большего свободного места до препятствия.</p>

		
3	<p>Найденный путь имеет постоянный размер.</p>	<p>Найденная трасса может быть переменного размера.</p>
4	<p>Нахождение единственного пути в 2D.</p> 	<p>Нахождение нескольких трасс между уже размещенными объектами в 2D и 3D.</p> 
5	<p>Сглаживание пути с помощью R-min (используются кривые линии и дуги).</p> 	<p>Сглаживание пути с помощью R-min (используются окружности и дуги).</p> 

Важнейшим вопросом использование рецепторных геометрических моделей являются вопросы точности этой модели. Очевидно, что если моделируемый объект набирается как мозаика из кубиков, то чем меньше размер этих кубиков, тем точнее представляется ими моделируемый объект. Расплачиваться за увеличение точности приходится затратами вычислительных ресурсов. Оценка геометрической точности представления модели приведена в диссертации Ситу Лина [169]. Проведенное нами исследование прохождения трассы на тестовых примерах с частным заранее известным результатом для 2-х проведенных тестов представлена на рисунке 4.35. На этих графиках приведены вычисленные значения погрешностей по осям Ox , Oy и Oz соответственно.



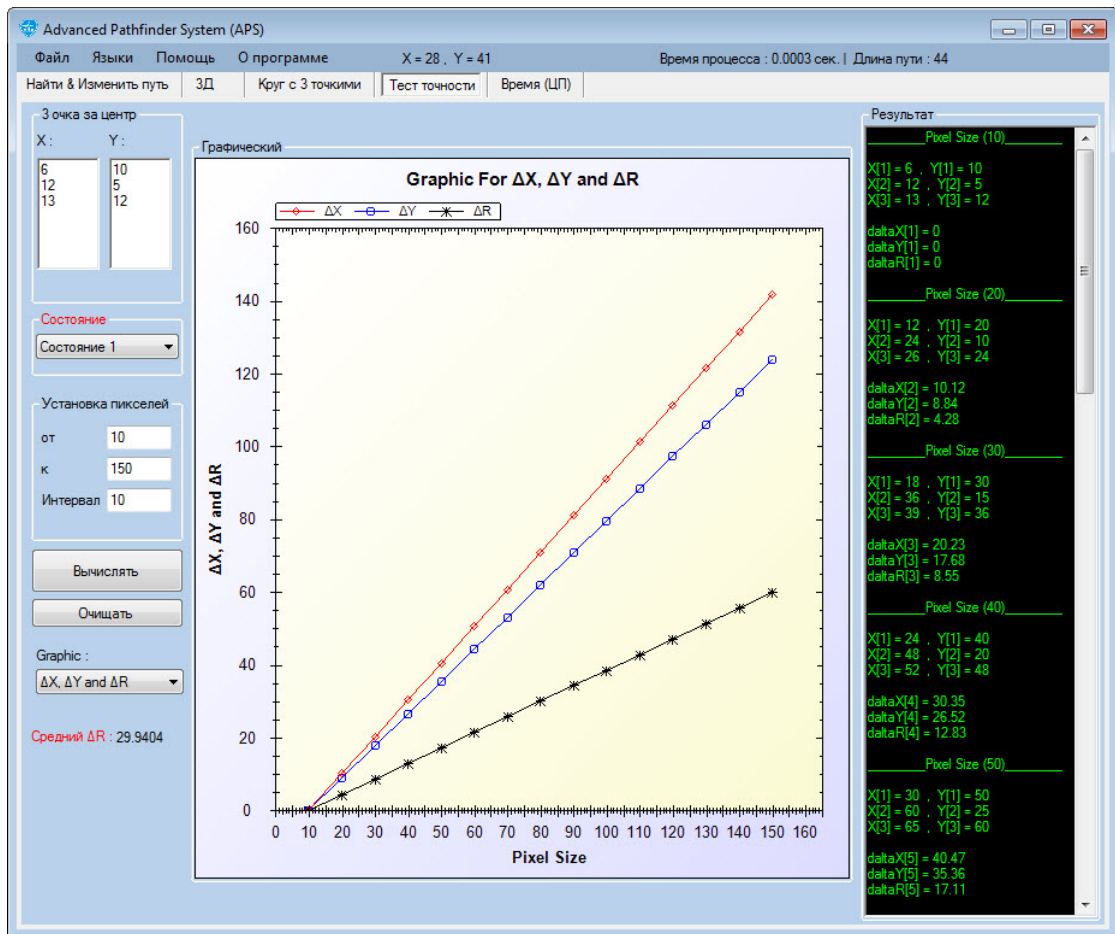


Рисунок 4.35 – Графики оценки точности рецепторной моделей, полученных на основании тестовых примеров с заранее известным результатом

Оценка производительности предложенного алгоритма зависит прежде всего от производительности конкретного компьютера, на котором проводились замеры скорости (частоты и архитектуры процессора, количества его ядер, количества и латентности оперативной памяти и производительности жесткого диска). Результаты оценки производительности для тестовых примеров применены для компьютера мощности (на данный момент) чуть выше средней. Результаты теста скорости приведены на рисунке 4.36.

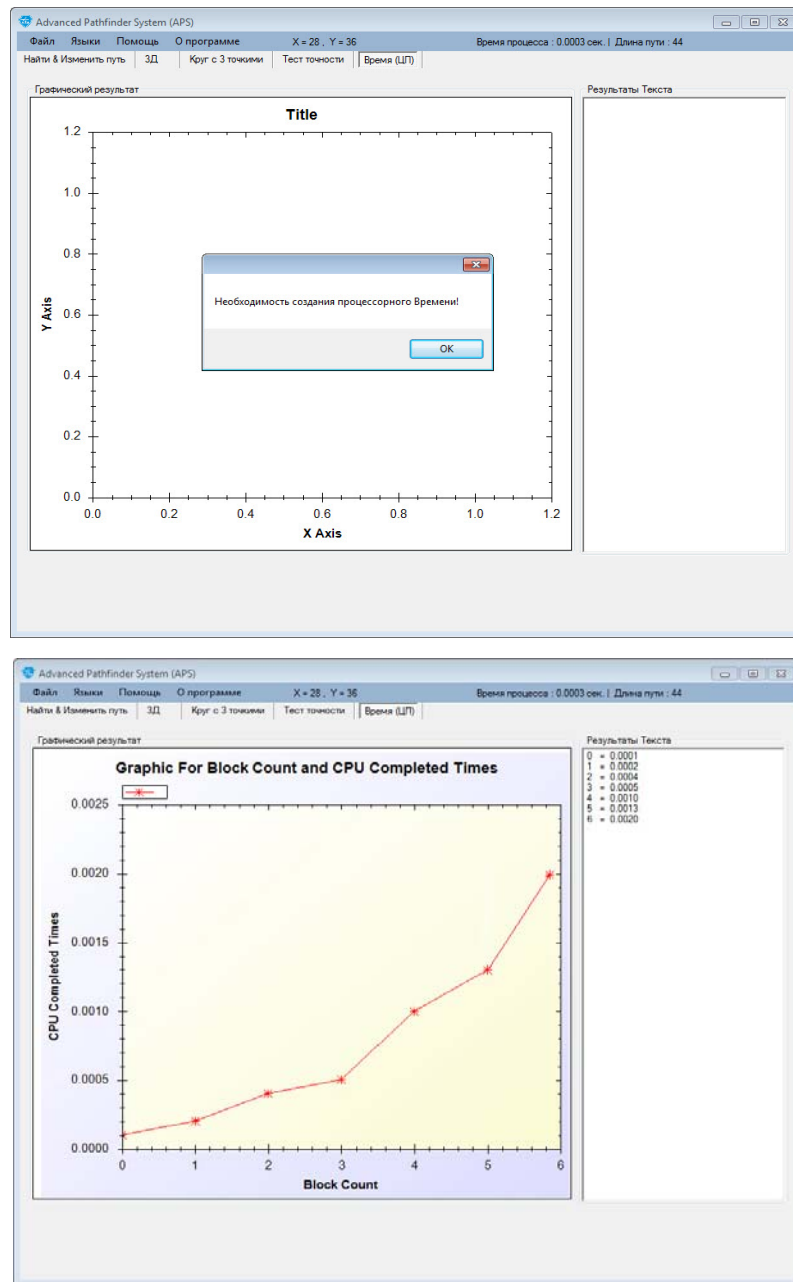


Рисунок 4.36 – Результаты теста скорости для конкретного примера компоновки

На рисунке 4.35 график показывает необходимое процессорное время прокладки трассы в тестовом примере в зависимости от размера рецептора. Из этого рисунка видно, что именно процессорное время при увеличении дискретности пространства увеличивается примерно по квадратичной параболе, но все равно составляет доли секунды.

Важнейшим вопросом верификации разработанных алгоритмов проектирования телесной трассы является оценка геометрической точности рецепторной модели. Она также проводилась с использованием геометрической модели из композиции примитивов, в которой идеальное прохождение трассы было заранее просчитано, что позволяло получить эталонные значения формы трассы. Расчетные же данные, полученные с использованием предложенного алгоритма, позволяют оценить точность модели как погрешность формы между идеальной геометрической моделью трассы и реальным прохождением телесной трассы, построенных с использованием разработанных нами алгоритмов и геометрических моделей. Результаты подсчета отклонений геометрической формы, рассчитанные при использовании тестовых моделей в рецепторной матрице размерами $1\text{м} \times 1\text{м} \times 1\text{м}$, представлены на рисунке 4.37.

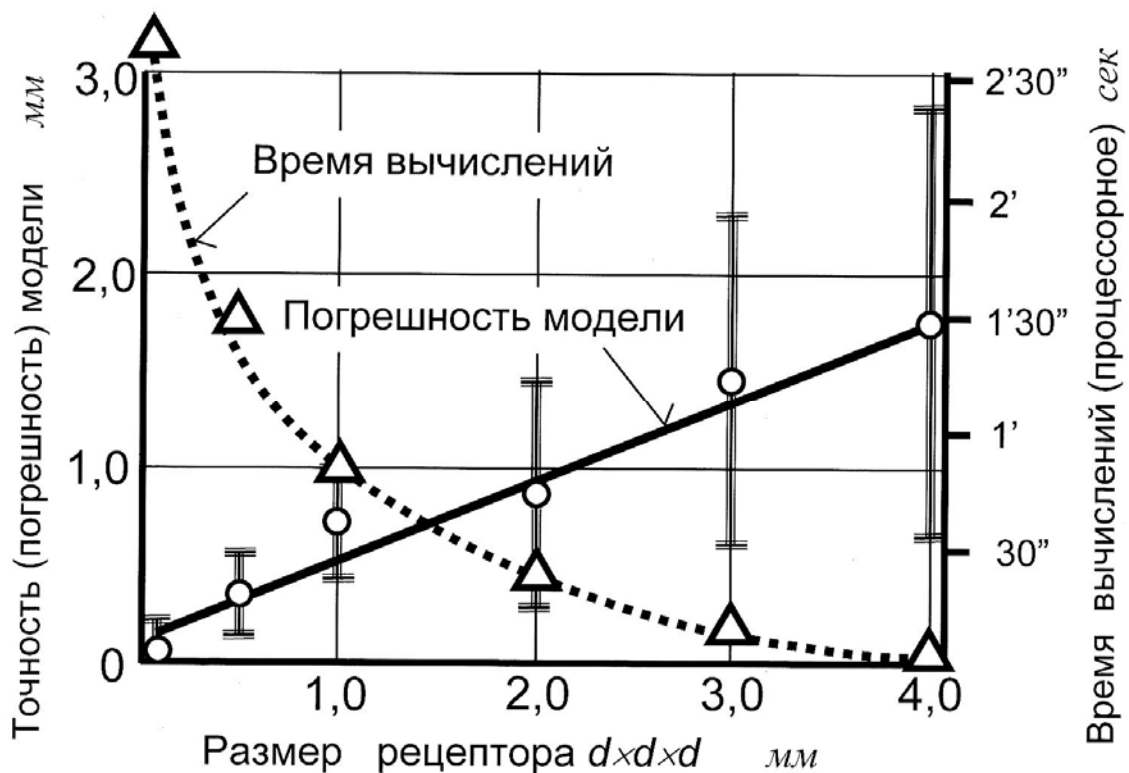
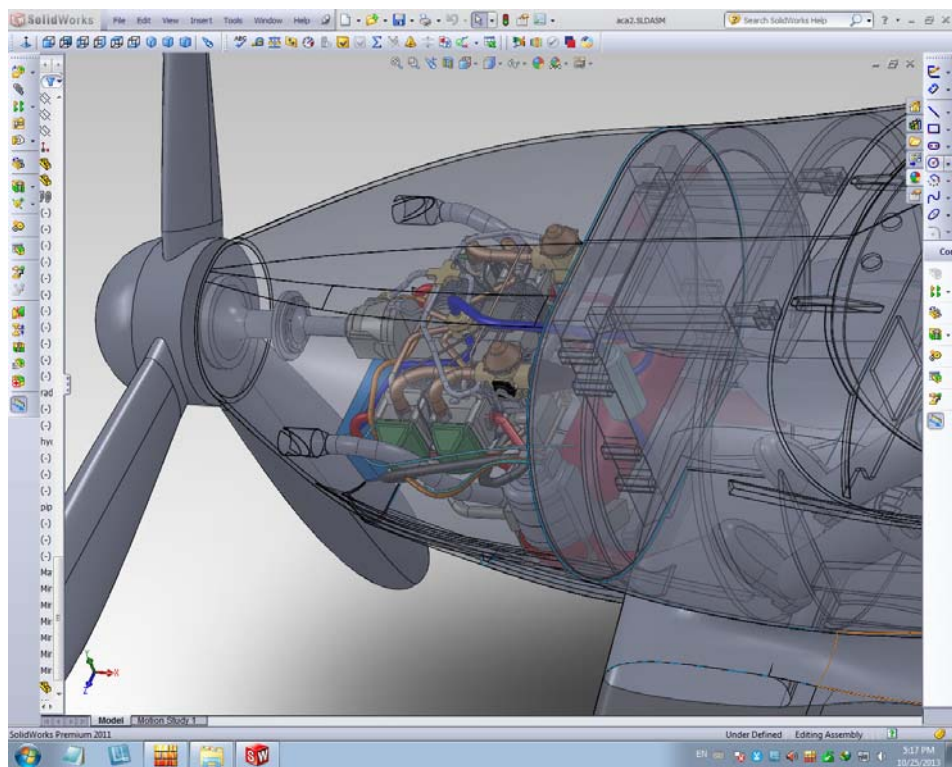
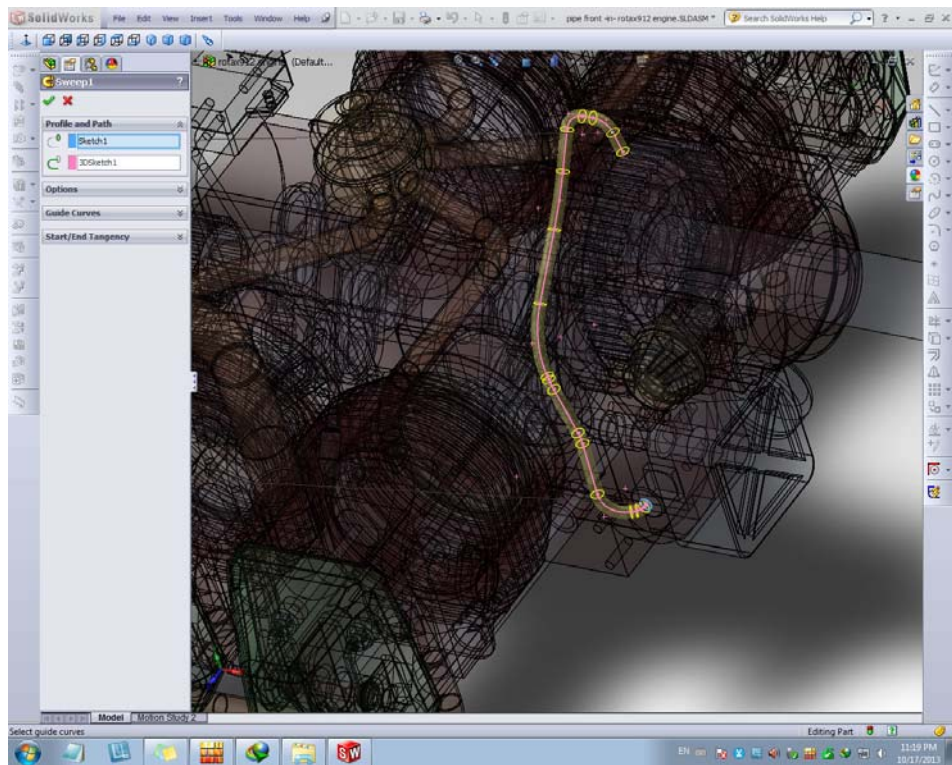


Рисунок 4.37 – Расчетные характеристики точности рецепторной геометрической модели

Анализ результатов этого исследования (см. рисунок 4.37) показывает, что точность описания геометрической формы ожидаемо зависит от размера рецептора (в исследовании анализировались рецепторы формы правильного куба со стороной d), т. е. с увеличением d погрешность описания формы растет по примерно линейной зависимости и ее математическое ожидание составляет примерно $0,9 d \pm 0,28d$ при доверительном интервале $\pm 3\sigma$. Имитационное моделирование позволило оценить и затраты процессорного времени, которые также ожидаемого растут при уменьшении размера рецептора d и соответствующим увеличением точности модели по параболической зависимости от нескольких секунд до почти 3-х минут при использовании персонального компьютера мощности, несколько выше средней для современного уровня. Это означает, что при размере рецептора, например, 0,2 мм погрешность представления формы составляет $0,18 \pm 0,06$ мм, что вполне достаточно для этапа проектирования новой трассы и является разумным компромиссом между ожидаемой точностью алгоритма и затратами вычислительных ресурсов для его реализации.

Необходимо отметить, что имеются немалые резервы для снижения и требуемой вычислительной мощности при реализации рецепторных геометрических моделей (например путем использования адаптивных размеров рецептора), однако рассмотрение этого вопроса выходит за рамки настоящей статьи. Быстрый рост производительности вычислительной техники делает дискретные рецепторные модели все более привлекательными и все более востребованными в практике проектирования высокотехнологичной техники.

Примеры практического использования предложенного алгоритма и программного обеспечения при разработке проектных решений модификации легкого самолета “АСА-2” приведены на рисунках 4.38. Использование рецепторного метода наиболее эффективно именно при модификации существующих компоновок, что показано на примере легкого самолета “АСА-2”.



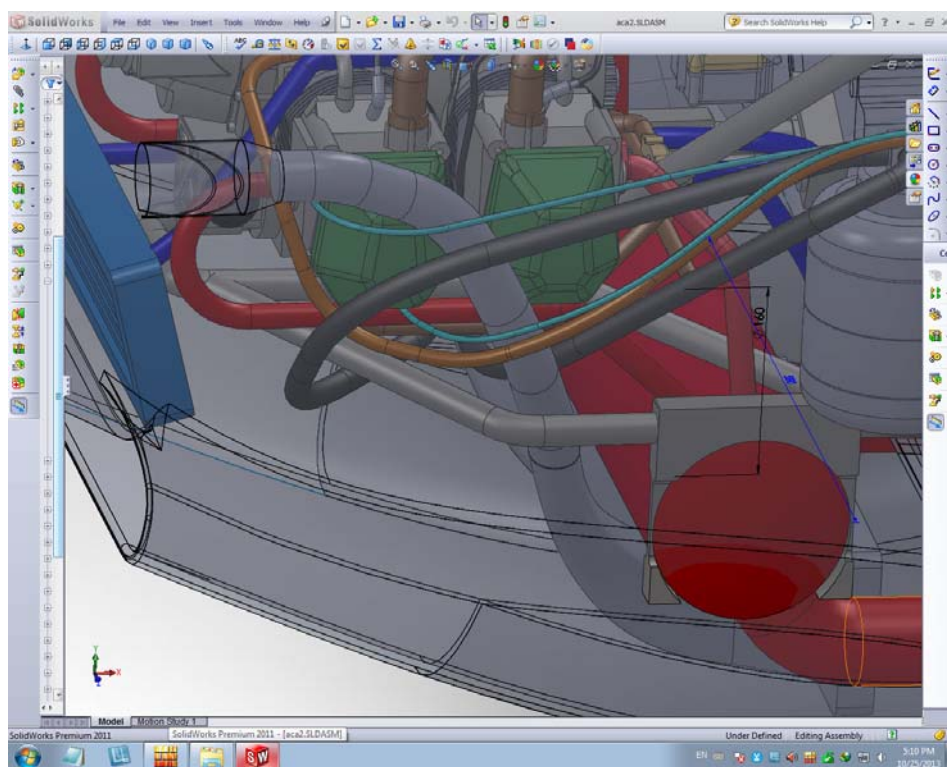
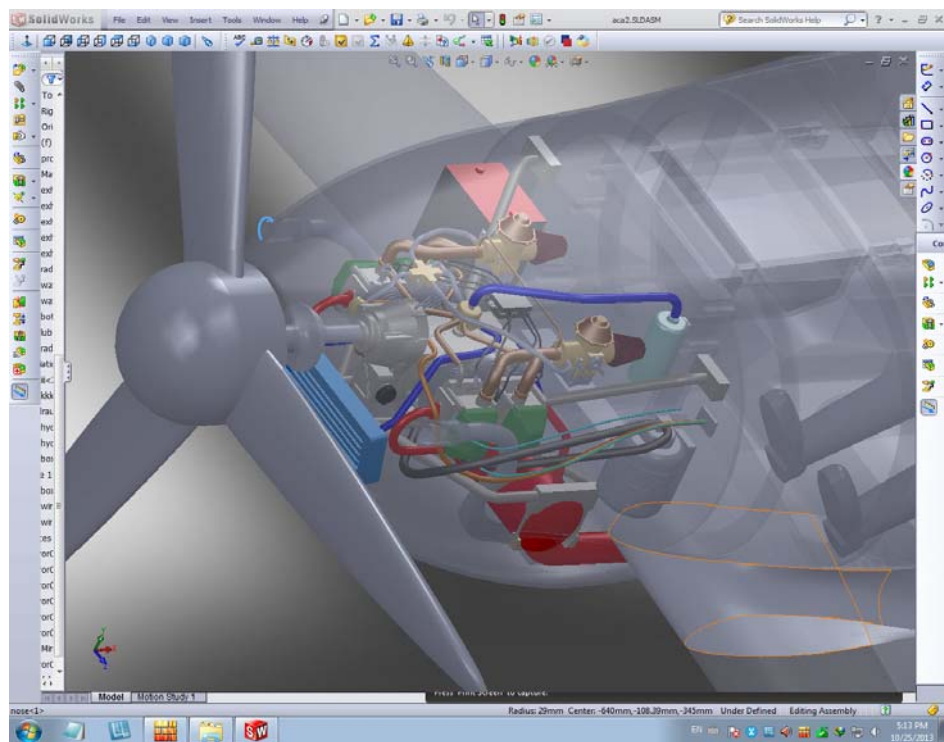


Рисунок 4.38 – Использование предложенного метода при компоновке легкого самолета “АСА-2”

Следует признать, что описанная программа, реализующая рецепторный метод телесной трассировки, не интегрирована в какую-либо существующую САД-систему, поэтому ввод информации о компоуемых объектов в виде композиции примитивов вводится в программу в параметрическом виде. Это снижает производительность процесса компоновки. Однако такая интеграция с САД-системой, для которой данная программа является встроенным расчетным модулем, является самостоятельной научной проблемой и выходит за рамки настоящего исследования.

Для оценки эффективности практического использования разработанной в рамках диссертации геометрической модели и реализующего ее математического и программного обеспечения также использовался метод имитационного моделирования определенной тестовой компоновочной ситуации в рецепторной матрице размерностью $1\text{м} \times 1\text{м} \times 1\text{м}$. В сцене, с теми же размерами $1\text{м} \times 1\text{м} \times 1\text{м}$, в которой были размещены несколько уже вроде бы скомпонованных объектов (примитивов и композиций примитивов), подготовленному пользователю предлагалось с использованием конкретной САД-системы (использовалась SolidWorks 2012) смоделировать трассу заданного сечения между двумя точками (начальной и конечной) с заданными координатами. Такая же задача ставилась и при использовании разработанного алгоритма. В последнем случае в рецепторную матрицу была уже введена информация в размерах и форме уже скомпонованных объектов, а также параметрах проектируемой трассы (радиусе сечения, минимально допустимом радиусе кривизны и координатах конечных точек). Результаты такого “соревнования” представлены на рисунке 4.39. Видно, что время работы разработанного алгоритма определяется прежде всего дискретностью рецепторной матрицы d и в меньшей степени – количеством скомпонованных объектов, через которые предстоит простроить трассу. Время же построения трассы конкретным исполнителем существенно зависит от количества объектов-препятствий и в любом случае ока-

зывает существенно большим (до примерно 300 раз), чем время построения трассы алгоритмом в автоматическом режиме.

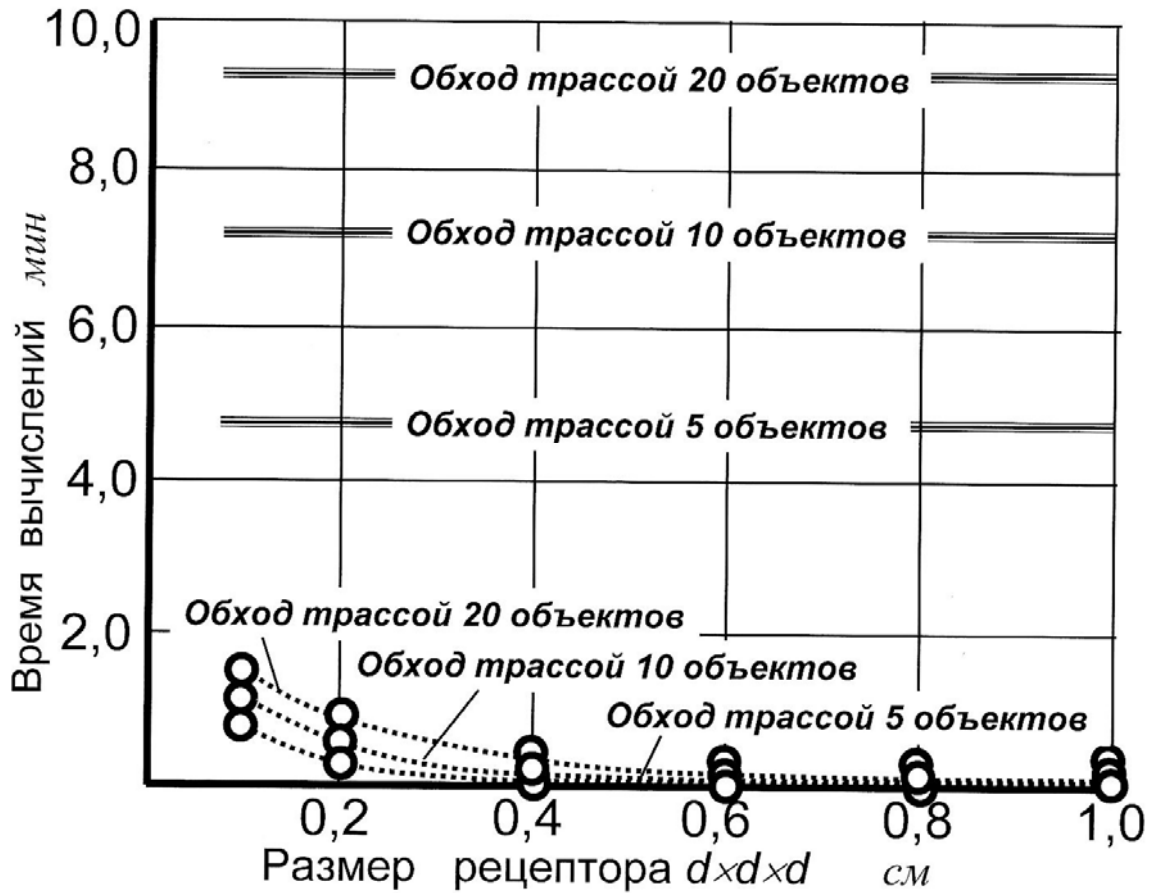


Рисунок 4.39 – Результаты имитационного моделирования затрат времени на построение трассы в пользователем интерактивном режиме и алгоритмом в автоматическом

Хотя рост скорости в десятки и сотни раз (в зависимости от требуемой точности формирования модели) для разработанного нами алгоритма налицо, справедливости ради необходимо признать, что по скорости сравнивается лишь сам процесс формирования трассы (т.е. его расчетная часть), оставляя за кадром другие временные затраты – ввод информации в рецепторную модель и вывод проектной информации, который для имеющегося программного комплекса будет

несколько больше, чем при использовании САД-системы. Кроме того, полученный при этом результат будет не таким наглядным, как при использовании САД-системы. К тому же, приведенные на рисунке 4.39 данные будут зависеть от квалификации конкретного пользователя и степени его владения САД-системой, что позволит ему выполнить построение заданной трассы либо быстрее, либо медленнее.

Но и у разработанного нами алгоритма есть и скрытые преимущества – его использование гарантирует обеспечение заданные проектных характеристик, прежде всего минимального радиуса кривизны. Использование же САД-системы пользователем в интерактивном режиме оставляет этот вопрос за кадром, обеспечивая лишь визуальный контроль плавности трассы. Если же при этом возникнет вопрос расчета конкретного значения радиуса кривизны, то пользователю придется средствами САД-системы определять координаты дискретных точек трассы и проводить расчеты кривизны, то выигрыш разработанного алгоритма в скорости работы будет подавляющим.

Результаты данного исследования также использованы в учебном процессе кафедры “Инженерная графика” Московского авиационного института (Национального исследовательского университета) в курсах для Центра повышения квалификации преподавателей МАИ “Методы геометрического моделирования и автоматизации жизненного цикла в САД/САМ/САЕ” и курса для аспирантов МАИ “Методы геометрического моделирования и автоматизации воспроизведения изделий в САД/САМ/САЕ”.

ЗАКЛЮЧЕНИЕ И ВЫВОДЫ ПО РАЗДЕЛУ 4

Проведенный анализ и исследование предложенного алгоритма и реализующей его программы телесной трассировки, а также его применение в разработке компоновки легкого самолета позволило сделать следующие выводы:

1. Проведенная детализация структуры разработанного алгоритма обеспечивает решение задачи телесной трассировки с учетом плавности и заданной площади сечения трассы.

2. Для предложенного алгоритма разработано реализующее его программное обеспечение на языке C#, обеспечивающего средствами интерфейса программы настройку режимов и параметров трассировки, а также визуализацию полученных компоновочных решений.

3. Модификации предложенного алгоритма трассировки и оптимизация его информационной структуры позволила увеличить быстродействие программы по сравнению с ближайшими аналогами (алгоритмами Дейкстры и A*) в 300 -1200 раз.

4. Проведена оценка точности представления телесной трассы рецепторной матрицей, показавшей на тестовых примерах, что погрешность представления зависит от размера рецептора d и составляет примерно $0,9 d \pm 0,28d$ при доверительном интервале $\pm 3\sigma$.

5. Проведенная оценка производительности реализации предложенного рецепторного алгоритма показала, что процессорное время расчета компоновки возрастает примерно по параболической зависимости от количества рецепторов на единице длины рецепторной матрицы и в тестовых примерах составляет от 0,05 до 1.8 секунды.

6. Проведенное с помощью предложенного метода и обеспечивающего его программного обеспечения трассировки позволило провести исследование возможности прокладки воздуховода в моторном отсеке легкого самолета "АСА-2", что выявило возможности модификации моторного отсека этого самолета. Результаты диссертационного исследования также внедрены в учебный процесс кафедры инженерной графики МАИ.

7. Программа, реализующая предложенный рецепторный метод телесной трассировки, не интегрирована в какую-либо существующую САД-систему, что

снижает производительность процесса компоновки, и тем самым, эффективность ее использования. Однако такая интеграция с САД-системой, для которой данная программа является встроенным расчетным модулем, является самостоятельной научной проблемой и выходит за рамки настоящего исследования.

8. Имитационное моделирование процесса проектирования трассы показало, что использование разработанной геометрической модели, алгоритма и программного обеспечения увеличивает на расчетном этапе скорость проектирования каналовой поверхности по сравнению с использованием САД-системы в интерактивном режиме в десятки и сотни раз (в зависимости от заданной точности), обеспечивая при этом заданные параметры проектирования (прежде всего минимальный радиус кривизны каналовой поверхности).

ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ ПО РАБОТЕ

1. Исследование физической и математической постановки задачи автоматизированной телесной трассировки позволило сформулировать ее как многокритериальную задачу математического программирования.

2. Показано, что использование для решения поставленной задачи рецепторного метода геометрического моделирования позволяет разрабатывать новые подходы и геометрические модели для компоновки таких сложных по своей геометрической форме объектов как каналовые поверхности.

3. Показана невозможность использования даже лучших известных алгоритмов дискретной трассировки (алгоритма Дейкстры и алгоритма Масатомо Канехары A^*) для автоматизации трассировки каналовых поверхностей.

4. Разработана геометрическая модель и алгоритм построения главной направляющей линии каналовой поверхности для плоской и пространственной трассы, являющийся глубокой модернизацией алгоритма A^* и устраняющий основные

ограничения прототипа A^* - возможность прокладки плавных трасс на заданных расстояниях от областей запрета.

5. Разработанные в рамках данного исследования эвристики, направленные на выбор рациональных направлений движений к следующей точке будущей траектории по 8-ми смежным вершинам (на плоскости) и по 26-и смежным вершинам в пространстве увеличивает скорость работы алгоритма трассировки в 3..5 раз.

6. Для предложенного алгоритма разработано реализующее его программное обеспечение на языке C#, обеспечивающего средствами интерфейса программы настройку режимов и параметров трассировки, а также визуализацию полученных компоновочных решений.

7. Модификации предложенного алгоритма трассировки и оптимизация его информационной структуры позволила увеличить быстродействие программы по сравнению с ближайшими аналогами (алгоритмами Дейкстры и A^*) в 300 -1200 раз.

8. Проведенная оценка точности представления телесной трассы рецепторной матрицей показала на тестовых примерах, что погрешность представления зависит от размера рецептора d и составляет примерно $0,9 d \pm 0,28d$ при доверительном интервале $\pm 3\sigma$.

9. Проведенная оценка производительности реализации предложенного рецепторного алгоритма показала, что процессорное время расчета компоновки возрастает примерно по параболической зависимости от количества рецепторов на единице длины рецепторной матрицы и в тестовых примерах составляет от 0,05 до 1.8 секунды.

10. Имитационное моделирование процесса проектирования трассы показало, что использование разработанной геометрической модели, алгоритма и программного обеспечения увеличивает на расчетном этапе скорость проектирования каналовой поверхности по сравнению с использованием САД-системы в интерактивном режиме в десятки и сотни раз (в зависимости от заданной точности), обес-

печивая при этом заданные параметры проектирования (прежде всего минимальный радиус кривизны каналовой поверхности).

11. Проведенное с помощью предложенного метода и обеспечивающего его программного обеспечения трассировки позволило провести исследование возможности прокладки воздуховода в моторном отсеке легкого самолета “АСА-2”, что выявило возможности модификации моторного отсека этого самолета. Результаты диссертационного исследования также внедрены в учебный процесс кафедры инженерной графики МАИ.

СПИСОК ЛИТЕРАТУРЫ

1. Абрайтис Л.Б., Шейнаускас Р.И., Жилевичюс В.А. Автоматизация проектирования ЭВМ. Автоматизированное техническое проектирование конструктивных узлов цифровых устройств. – М. : Советское радио, 1978. - 272 с.
2. Аведьян А.Б., Бибииков С.Ю., Маркин Л.В. и др. Компоновка самолетов /Под ред. М.Ю.Куприкова.-М.: Изд-во МАИ, 2012. -296 с.
3. Александрович В.П. Инженерный способ конструирования циклических поверхностей и его приложения. Автореф. дисс. . канд. техн. наук. М., МАИ, 1990.: 16 с.
4. Аминов Ю.А. Дифференциальная геометрия и топология кривых. -М.: Наука, 1987. 159 с.
5. Андреев В.А. Разработка и исследование машинных методов проектирование каналовых поверхностей по заданным условиям. Автореф. дисс. . канд. техн. наук. -М., МАТИ, 1975.: 25 с.
6. Андреев В.А., Зворыкин В.Н., Коноров Л.В. и др. Расчёт и построение контуров самолёта на плазе. - М., Оборонгиз, 1960. - 492 с.
7. Арайс Е. А., Дмитриев В. М. Автоматизация моделирования многосвязных механических систем. - М.: Машиностроение, 1987. -240 с.
8. Аристова И.В. Методы и алгоритмы приближенного решения комплексной задачи компоновки. Автореф. канд. дисс. на соиск. уч. степ. канд. техн. наук. Харьков, ХИРЭ, 1984, - 26 с.
9. Аристова И.В. Об автоматизации процесса компоновки машинного зала энергоблока // Проблемы машиностроения, 1982, № 15, с. 68-71.
10. Артамонов Б.Л. Завалов О.А., Маслов А.Д., Мойзых Е.И. Разработка виртуальных параметрических моделей типовых элементов конструкции вертолета

- // Электронный журнал “Прикладная геометрия”, вып. 11, № 22 (2009) , с. 89-103.
11. Бабаков В.В. Проектирование поверхностей кривыми второго порядка в самолётостроении. — М., Машиностроение, 1969.: 124 с.
 12. Базилевич Р.П. Декомпозиционные и топологические методы автоматизированного метода конструирования электронных устройств. — Львов.: Вища школа. — 1981. — 168 с.
 13. Базилевич Р.П. Обобщенный подход к формализации задачи машинной трассировки межсоединений на плоскости // Изв. ВУЗов СССР. Радиоэлектроника, 1974, № 6, С. 98-103.
 14. Безкоровайный В. П. Разработка методов оптимизации трасс магистральных газопроводов и их разветвлений для сетей произвольной конфигурации: Автореферат диссертации на соискание ученой степени канд. техн. наук.— М., 1978. - 25 с.
 15. Безкоровайный В.П., Бородавкин П.П., Андреев О.П. Автоматизированное проектирование газотранспортных систем. - М.: Недра, 1990. - 176 с.
 16. Блюк А.В. Графоаналитическое конструирование поверхностей каналового типа по наперед заданным площадям поперечных сечений. Автореф. дисс. канд. техн. наук. Киев, КПИ; 1970.: 22 с.
 17. Богацкий И.З. Графоаналитическое конструирование выхлопного патрубка двигателя. // Прикладная геометрия и инженерная графика. - К., Будвельник, 1979.: вып. 28, с. 60 - 62.
 18. Брысин В.А., Филиппов П.В. Моделирование каналовых поверхностей с параболическими поперечными сечениями. // Прикладная геометрия и инженерная графика. К., Будвельник,, 1992.: вып. 53, с. 23 - 25.
 19. Вальков К.И. Основы геометрического моделирования. -Л.: ЛИСИ, 1986.

20. Василевский О.В. Геометрическое моделирование поверхностей каналов и автоматизация их проектирования. Автореф. дисс. . канд. техн. наук. -Киев, КИСИ, 1982.: 17 с.
21. Вермишев Ю.Х. Методы автоматизированного поиска решений при проектировании сложных технических систем. – М.: Радио и связь, 1982 – 152 с.
22. Вермишев Ю.Х. Основы автоматизации проектирования. – М.: Радио и связь, 1988 – 280 с.
23. Волков Е.А. Численные методы. М.: Наука, 1987. - 248с.
24. Волошин В.В. Автоматизация проектирования летательных аппаратов. - М.: Машиностроение, 1991. – 256 с.
25. Гаврилов В.Н. Автоматизированная компоновка приборных отсеков летательных аппаратов. – М.: Машиностроение, 1988. – 136с.
26. Гардан И., Люка М., Машинная графика и автоматизация конструирования. Пер. с франц.– Мир, 1987. – 270 с.
27. Герасименко Е.П., Зозулевич Д.М. Минимизация вычислительного процесса при решении геометрических задач с помощью рецепторных матриц. – В кн.: ВТ в машиностроении. Минск, ИТК АН БССР: июнь, 1970. – с. 26 – 40 .
28. Герасименко Е.П., Зозулевич Д.М. Методы формирования трехмерных рецепторных матриц на ЭВМ. // В кн.: Вычислительная техника в машиностроении. – Минск. ИТК АН БССР, 1971. – с. 24 – 31 .
29. Герасименко Е.П., Кот В.И. и др. Автоматизация проектирования печатных блоков с модулями произвольной формы. – М.: Машиностроение , 1979. – 167 с.
30. Гиллой В. Интерактивная машинная графика. Структуры данных, алгоритмы, языки. Пер. с англ. – М.: Мир, 1981. – 384 с.
31. Горелик А.Г. Автоматизация инженерно-графических работ с помощью ЭВМ. – Минск: Высшая школа, 1980 г. – 206 с.

32. Горелик А.Г. Об алгоритмическом построении теоретико-множественного описания геометрического объекта. // Вычислительная техника в машиностроении. – Минск: ИТК АН БССР, 1968, февраль, - с. 36 - 49 .
33. Горелик А.Г. Методы геометрического моделирования при автоматизированном проектировании объектов сложной структуры. Автореф. дисс. на соискание уч. степени д.т.н. –Минск: ИТК АН БССР, 1983.- 46 с.
34. Горелик А.Г. Автоматизация инженерно-графических работ с помощью ЭВМ. – Минск: Высшая школа, 1980 г. – 206 с.
35. Грибов С.М. Дискретная геометрия интерактивного конструирования кинематических поверхностей на основе конечных сумм. Автореф. дисс. . доктора техн. наук. Киев, КНУБА, 1994.: 37 с.
36. Григорьев С.Н., Локтев М.А., Толок А.В. Построение воксельных моделей геометрических объектов // Прикладная Информатика. 2013. № 4. С. 50-55.
37. Грувер М., Зиммерс Э. САПР и автоматизация производства. Пер. с англ. - М.:Мир, 1987. – 528 с.
38. Давыдов Ю.В., Злыгарев В.А. Геометрия крыла: Методы и алгоритмы проектирования несущих поверхностей. М.: Машиностроение, 1987. - 131 с.
39. Денискин Ю.И. Обобщённые методы геометрического моделирования объектов и управления их формой при параметрическом представлении. Автореф. дисс. доктора техн. наук. — М., МГАИ, 2000.: 38с.
40. Денискин Ю.И., Егоров Э.В., Нартова Л.Г., Куприков М.Ю. Прикладная геометрия. Научные основания и применение в технике. – М.: Изд-во МАИ-Принт, 2010.
41. Дёмина В.А. Геометрическое моделирование функционально зависимых каналовых поверхностей. Автореф. дисс. канд. техн. наук. - М., МТИПП, 1991.: 17 с.

42. Дорошенко Ю.А. Політканинні перетворення як апарат деформативного конструювання каналових поверхностей. // Прикладная геометрия и инженерная графика. К., КДТУБА, 1997.: вып. 61с. 84 - 87.
43. Дорошенко Ю.А. Геометрическое моделирование трубопроводов из унифицированных элементов. // Прикладная геометрия и инженерная графика. К., БудІвельник, 1988.: вып. 45, с. 92 - 94;
44. Драганов Б.Х., Карпусь А.Т., Мищенко А.В. Метод построения каналов двигателей внутреннего сгорания на основе свойств гармонических функций. // Прикладная геометрия и инженерная графика. К., БудІвельник,, 1979.: вып. 28, с. 40 - 42.
45. Драганов Б.Х., Круглов М.Г. Обухова В.С. Конструирование впускных и выпускных каналов двигателей внутреннего сгорания. Киев, Вища школа. 1987.: 175 с.
46. Егер С.М., Лисейцев Н.К., Самойлович О.С. Основы автоматизированного проектирования самолетов. – М.: Машиностроение , 1986. – 232 с.
47. Егоров Э.В., Тузов А.Д. Моделирование поверхностей агрегатов ЛА - М.: Изд-во МАИ, 1988.- 88с.
48. Егоров Э.В., Нартова Л.Г. Конструктивная геометрия. – М.: Изд-во МАИЮ 2012. – 160 с.
49. Есмуханова Ж. Ж. Геометрические методы расчета конфигурации инженерных сетей. Автореф. дис. канд. техн. наук: 05.01.01.— М.: МАИ, 1987.— 17 с.
50. Есмуханова Ж. Ж. Алгоритм построения оптимальной конфигурации инженерной сети // Моделирование задач науки и техники методами начертательной геометрии.— Алма-Ата: изд. КазПТИ.— 1986.— С. 43- 46.
51. Есмуханова Ж. Ж. Геометрические задачи на инженерных сетях // Прикладная геометрия и инженерная графика в теории и практике авиационного автоматизированного проектирования.— Киев: КИИГА, 1984.—С. 58—62.

52. Есуханов Ж. М., Есмуханова Ж. Ж. Подсчёт числа разомкнутых связывающих линий // Материалы XVI научно-технической конференции профессорско-преподавательского состава Казахского политехнического института имени В. И. Ленина.— Алма-Ата: КазПТИ, 1982.
53. Зверев И.И. Проектирование гидравлических систем летательных аппаратов. —М.: ЕЕМедиа, 2012. – 149 с.
54. Зелев В.П. Исследование машинных методов проектирования и расчёта канальных поверхностей сложных технических форм. Автореф. дисс.— канд. техн. наук. М., МАТИ, 1977.: 18 с.
55. Зиман Ю.Л., Гринберг Г.С. Некоторые новые возможности волнового алгоритма М.: ИТМ и ВТ АН СССР, 1973 г. -22 с.
56. Зиман Ю.Л., Рябов Г.Г. Волновой алгоритм и электрические соединения // В кн. : Электронные вычислительные машины. М.: ИТМ и ВТ АН СССР, 1965 г. С. 47-62.
57. Зозулевич Д. М., Шерлинг Д.Р. Методы реализации на ЭЦВМ теоретико-множественных операций над плоскими многосвязанными областями. // В кн.: Вычислительная техника в машиностроении. Минск, изд. НТК АН БССР, 1969, с. 26-35.
58. Зозулевич Д. М., Максимова Л. Г. Выполнение на ЭЦВМ некоторых операций с трехмерными кусочно-заданными объектами. // В кн.: Вычислительная техника в машиностроении. Минск, изд. НТК АН БССР, 1970, с. 75-84.
59. Зозулевич Д. М., Ловчев Э. М. Построение с помощью ЭЦВМ изображений трехмерных объектов, заданных кусочно-аналитическими моделями. // В кн.: Вычислительная техника в машиностроении. Минск, изд. НТК АН БССР, 1971, с. 64-76.
60. Зозулевич Д.М. Машинная графика в автоматизированном проектировании. — М.: Машиностроение, 1976. – 240 с.

61. Иванов Г.С. Конструирование технических поверхностей. – М.: Машиностроение, 1987. – 192 с.
62. Иванов Г.С. Поверхности и кривые расслояемых нелинейных преобразований в начертательной геометрии и технике. Автореф. дисс. . доктора техн. наук. М., МИСИ, 1977.: 32 с.
63. Иванов Г.С. Теоретические основы начертательной геометрии: Учебное пособие. - М.: Машиностроение, 1998. — 158 с.
64. Иванов Г. С., Есмуханова Ж. Ж. Построения кратчайших связывающих сетей на топографической поверхности // Начертательная геометрия и машинная графика в практике решения инженерных задач.— Омск: ОмПИ. 1987.—С. 27- 32.
65. Иванов Г.С., Миролубова Т.И. Теоретические предпосылки конструирования осей трубопроводов как одномерных обводов второго порядка гладкости. // Прикладная геометрия и инженерная графика. К., КЛУБА, 2001.: вып. 68, с. 38- 441.
66. Идельчик И.Е. Справочник по гидравлически сопротивлением. Изд 3-е, перераб. и дополн. –М.: Машиностроение, 1992. – 872 с.
67. Исакова С.Д. Исследование поверхностей зависимых сечений методом нелинейных преобразований применительно к проектированию поверхностей трубопроводов. Автореф. дисс. канд. техн. наук. -М., МАДИ,1977.: 16 с.
68. Калинин Б.В. Автоматизация синтеза топологии и размещения коммуникационных сетей. Автореф. канд. дисс. на соиск. уч. степ. канд. техн. наук. Куйбышев, КуАИ, 1985, - 16 с.
69. Калинин Б.В., Цыбатов В.А. Об одном подходе к оптимизации размещения измерительно-вычислительных комплексов // Автоматизация научных исследований, Куйбышев: КуАИ, 1984, с. 41-47.

70. Клишин В.В. Моделирование трехмерных объектов на основе объемных базовых элементов формы. – Автореф. канд. дисс. на соиск. уч. степ. канд. техн. наук. М.: МЭИ, 1983, - 20 с.
71. Кожушко Н.А. К вопросу конструирования касающихся каналовых поверхностей. // Прикладная геометрия и инженерная графика. К., БудІвельник,, 1969.: вып. 9, с. 84- 89.
72. Компоновка самолетов / Под ред. М.Ю.Куприкова. – М.: Изд-во МАИ-ПРИНТ, 2012. – 262 с.
73. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. - Москва, Вильямс, 2005.
74. Корн Г.В. Методы формирования рецепторных геометрических моделей и их применение при решении инженерно-геометрических задач. Автореф. дисс. на соиск. уч. степ. канд. техн. наук. М.: МАДИ, 1990, - 22 с.
75. Корн Г.В. Применение рецепторных моделей при компоновке изделий авиационной техники. // В сб.: Тезисы докладов Всесоюзной научно-технической конференции. «Интегрированные системы автоматизированного проектирования». – М.: 1989. – с. 18-24.
76. Корн Г.В., Маркин Л.В. О возможностях математического обеспечения проектирования изделий машиностроения на базе рецепторных моделей. // В сб.: Материалы научно-технического семинара «Автоматизация проектирования передач и редукторов». – Ижевск, 1989. – с. 23 – 24.
77. Котов И.И. Каркасные поверхности зависимых сечений //Кибернетика графики и прикладная геометрия поверхностей. XI Тем. сб. научных трудов МАИ. 1974. - вып. 296 - с. 4-8.
78. Котов И.И. Методические основы и пути развития прикладной геометрии. // Кибернетика графики и прикладная геометрия поверхностей. М.: МАИ, 1970.: вып. 195, с. 39.
79. Котов И. И. Начертательная геометрия. - М: Высш. шк., 1970. - 384 с.

80. Котов И.И., Полозов В.С., Широкова Л.В. Алгоритмы машинной графики. М.: Машиностроение, 1977. – 231 с.
81. Коршиков С.Б., Падалко С.Н. Метод результирующего искажения параметров двухкомпонентной геометрической модели для обеспечения безопасности ее передачи по открытым каналам // Вестник Московского авиационного института, № 1, 2008, т.15, стр. 47-54.
82. Кулиш А.С. Конструирование циклических трубчатых поверхностей сопряжения. Автореф. дисс. . канд. техн. наук. М.: МТИПП, 1975.: 20 с.
83. Куприков М.Ю. Структурно-параметрический синтез облика самолета вертикального взлета и посадки Автореф. дисс. на соиск. уч. степ. д-ра техн. наук. – М.: МАИ. – 2000. – 32 с.
84. Курейчик В.М. Математическое обеспечение конструкторского и технологического проектирования с применением САПР. –М.: Радио и связь, 1990. – 352 с.
85. Ли К. Основы САПР (CAD/CAM/CAE). – СПб.: Питер, 2004. – 560 с.
86. Лелюшенко С.И. Исследование и управление дифференциально - геометрическими свойствами некоторых каналовых поверхностей типа «фюзеляж — мотогондола». Автореф. дисс. канд. техн. наук. -М., МАТИ, 1976.: 20 с.
87. Лузин С.Ю., Зудун С.В., Полубасов О.Б. Выбор конфигурации соединения компонентов электронных схем до этапа размещения // Технология и конструирование в электронной аппаратуре, 2001, № 4-5, С. 3-6.
88. Лузин С.Ю, Лячек Ю.Т., Петросян Г.С., Полубасов О.Б. Модели и алгоритмы автоматизированного проектирования радиоэлектронной и электронно - вычислительной аппаратуры. СПб.: БХВ-Петербург, 2010. — 224 с.
89. Лузин С.Ю., Лячек Ю.Т. Полубояров О.Б. Автоматизация проектирования печатных плат. Система топологической трассировки ТороR. – СПб, СПбГЭТУ “ЛЭТИ”, 2005, - 132 с.

90. Лузин С.Ю., Петросян Г.С., Полубасов О.Б. К вопросу о минимизации числа межслойных переходов при трассировке печатных плат // Технология и конструирование в электронной аппаратуре, 2009, № 3, с.13-15.
91. Лузин С.Ю., Полубасов О.Б. Проектирование печатных плат. Новые методы решения старых проблем // САПР и графика, 1997, № 11, С 58-59.
92. Лузин С.Ю., Полубасов О.Б. Топологический трассировщик печатных плат ТороR // Электронные компоненты, 2003, № 11, с. 59 – 62.
93. Лузин С.Ю., Попов С.И., Попов Ю.И. Автоматизация устранения клинчей в топологии печатного монтажа // Научно-технический вестник информационных технологий, механики и оптики, 2012, № 4 (80), с. 116-120.
94. Мальчевский В.В. Автоматизация процесса компоновки самолета: Учебное пособие для ФПК. – М.: МАИ, 1987. – 42 с.
95. Маркин Л.В. Геометрическое моделирование задач автоматизации размещения // Прикладная геометрия, инженерная графика, компьютерный дизайн, № 1 (11), 2007, с. 9-18.
96. Матвеевко А.М., Зверев И.И. Проектирование гидравлических систем летательных аппаратов. Учебник для ВУЗов. – М.: Машиностроение, 1982. – 296 с.
97. Математика и САПР: В 2-х кн. Кн. I. Пер. с франц./Шенен П., Гардан И. и др. - М.: Мир, 1988. - 204 с.
98. Мартынова О.Г. Геометрическое проектирование трубопроводных микросетей и воспроизведение их элементов. Автореф. дисс. канд. техн. наук.-М; МАИ, 1990.: 16 с.
99. Мезенцев Л.Г. Методы моделирования и компоновки каналовых поверхностей при автоматизации проектирования. Автореф. дисс. канд. техн. наук. Киев, КИСИ, 1985.: 19 с.

100. Медведский А.Л., Рабинский Л.Н., Курбатов А.С. Геометрическое моделирование авиационно-ракетных изделий // Электронный журнал “Прикладная геометрия”, вып. 11, № 22 (2009) , с. 79-87.
101. Миролубова Т.И.; Геометрическая модель переходного участка трубопровода конфузорно-диффузорного типа. // Электронный журнал «Прикладная геометрия», М.: МАИ (ГТУ) вып. 4, 2002.
102. Миролубова Т.И. Геометрические модели фасонных элементов однорукавных каналовых поверхностей Автореф. дисс. канд. техн. наук. -М.: МАИ; 2004.: 16 с.
103. В. Е. Михайленко, С. Н. Ковалёв, Н. И. Седлецкая и др. Инженерная геометрия с элементами теории параметризации - К.: УМК ВО, 1989.- 83 с.
104. В. Є. Михайленко, В. М. Найдиш, А. М. Підкоритов, І. А. Скидан. Інженерна та комп'ютерна графіка: Підручник; За ред. В. Є. Михайленка. - К.: Вища шк., 2000. – 342 с.
105. В. Є. Михайленко, М. Ф. Євстифеев, С. М. Ковальов, О. В. Кащенко. Нарисна геометрія - К.: Вища шк., 1993 - 271 с.
106. Мишин В.П., Осин М.И. Введение в машинное проектирование летательных аппаратов. - М.: Машиностроение, 1978. – 128 с.
107. Мухачева Э.А. Рациональный раскрой промышленных материалов. Применение АСУ. – М.: Машиностроение, 1984. – 176 с.
108. Наджаров К.М. Алгоритм и программа расчёта массово-инерционных параметров тел сложной переменной формы методом линейчатого шестигранника // Учебное пособие под ред. В.И.Якунина. М.: Изд. МАИ, 1982. - С. 27-42.
109. Наджаров К.М. Матричная стереометрия и геометрия масс тел переменной формы в проектировании самолёта: Автореф. дисс. докт. техн. наук. М.: 1983. - 34 с.
110. Наджаров К.М., Егоров Э.В., Якунин В.И. Теоретические основы формирования моделей поверхностей. – М.: МАИ, 1985. - 78 с.

111. Найдыш В.М. Конструирование поверхностей, проходящих через их специальные линии // Изв. высш. учеб. заведений. Авиац. техника, 1981, № 2, с. 88-90.
112. Найдыш В.М. Конструирование поверхностей с помощью нормальных отображений // В кн.: Начертательная геометрия и черчение. Алма-Ата, 1979, с. 69-73.
113. Найдыш В.М. Отображения поверхностей, учитывающие их дифференциально-геометрические характеристики // В кн.: Начертательная геометрия и черчение. Алма-Ата, 1979, с. 66-69.
114. Некрасова О.И. Геометрическое моделирование и автоматизация проектирования групп каналовых поверхностей. Автореф. дисс. канд. техн. наук. - М., МАИ, 1985.: 18 с.
115. Норенков И.П. Введение в автоматизированное проектирование технических устройств и систем –М.: Высшая школа, 1980. – 311 с.
116. Норенков И.П. Основы теории и проектирования САПР. –М.: Высшая школа, 1990. – 335 с.
117. Норенков И.П. Основы автоматизированного проектирования. –М.: Изд-во МГТУ им. Н. Э. Баумана, 2002. – 336 с.
118. Ньи Ньи Хтун. Улучшенный алгоритм трассировки пути, основанный на рецепторном методе // В сб.: Тез. докл. научно – практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2010 ». Тезисы докладов. – М.: Изд-во МАИ. – 2010. – С.108.
119. Ньи Ньи Хтун. Использование рецепторных моделей в задачах трассировки // В сб.: Тез. докл «Технологии Microsoft в теории и практике программирования: труды VII Всероссийской конференции студентов, аспирантов и молодых учёных». – М.: Изд-во Вузовская книга – МАИ. – 2010. – С.85 -86.

120. Ньи Ньи Хтун. Алгоритмы задач трассировки на основе рецепторных геометрических моделей // В сб.: Тез. докл. 9-я Международная конференция «Авиация и космонавтика – 2010 ». – М.: Изд-во МАИ. – 2010 . – С.314-315.
121. Ньи Ньи Хтун. Дискретные модели телесной трассировки // В сб.: Тез. докл. (Научно-практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2011 ») . – М.: Изд-во МЭЙЛЕР. С.190-191
122. Ньи Ньи Хтун. Использование рецепторного метода для проектирования каналовых поверхностей // В сб.: Тез. докл научно - практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2012 ». Тезисы докладов. – М.: Изд-во МАИ. – 2012. – С.256.
123. Ньи Ньи Хтун. Исследование рецепторного метода проектирования каналовых поверхностей в задачах компоновки авиатехники // В сб.: Тез. докл. 11-я Международная конференция «Авиация и космонавтика – 2012 ». – М.: Изд-во МАИ. – 2012 . – С.298-299.
124. Ньи Ньи Хтун. Использование рецепторных геометрических моделей в задачах телесной трассировки // В сб.: Тру. докл. «Международного конкурса научных работ по приоритетным направлениям развития науки, технологий и техники в Российской Федерации., МГТУ им. Н.Э.Баумана». – М.: Изд-во НИИ , 2012. С.146 – 147.
125. Образцов А.А. Топологические декомпозиционно-эвристические алгоритмы и комплекс программ оптимальной ресурсоэффективной компоновки химических производств. Автореф. дисс. канд. техн. наук.-М.: МРХТУ, 2009.: 20 с.
126. Обухова В.С. Конструктивно прикладная теория нелинейных осевых отображений и ассоциированных с ними алгебраических поверхностей. Автореф. дисс. доктора техн. наук. - Киев, КИСИ, 1991.: 41 с.

127. Обухова В.С., Василевский О.В. Применение метода сложения выпуклых кривых к конструированию каналových поверхностей. // Прикладная геометрия и инженерная графика. — К. БудІвельник, 1978.: вып. 26, с. 15- 17.
128. Осин М.И. Методы автоматизированного проектирования летательных аппаратов. - М.: Машиностроение, 1984. – 168 с.
129. Осипов В.А. Машинные методы проектирования непрерывно каркасных поверхностей. —М.;, Машиностроение, 1979.: 248 с.
130. Осипов В.А. Аппроксимация и интерполяция линий и поверхностей применительно к автоматизации, процессов проектирования и воспроизведения аэродинамических контуров самолёта. Автореф. дисс. доктора техн. наук. -М., МАИ; 1972.: 35 с.
131. Осипов В.А. Математическое моделирование в автоматизированной системе геометрических расчетов. // В кн.: Машинное проектирование, увязка и воспроизведение сложных деталей в авиастроении. - Иркутск, Изд. ИПИ, 1977, - с.4-14.
132. Осипов В.А, Проектирование непрерывных каркасов поверхностей с наперед заданными дифференциальными свойствами. // Научн. тр. МИРЭА, 1972, вып. 63. Инженерная графика. -с. 47-53.
133. Падалко С.Н. Определение тандемной модели как базовой формы представления многоуровневых математических моделей при проектировании аэрокосмической техники // Электронный журнал “Труды МАИ”, выпуск 71, 2013, URL: <http://www.mai.ru/science/trudy/published.php>
134. Пашенко О.Б. Компонировка оборудования маневренного самолета на базе матрично-топологического метода. Автореф. дисс. на соиск. уч. степ. канд. техн. наук. М.:, МАИ, 1990, - 22 с.
135. Пашенко О.Б. Автоматизированная компоновка оборудования в фюзеляже маневренного самолета // Известия ВУЗов. Авиационная техника, 1990, № 4. – с. 111-113.

136. Пашенко О.Б. Метод формирования компоновочной схемы радиоэлектронного оборудования на борту маневренного самолета // Известия ВУЗов. Авиационная техника, 1991, № 2. – с. 64-66.
137. Петренко А.И. Основы автоматизации проектирования. К.: Техника, 1982 - 295 с.
138. Петренко А.П., Тетельбаум А.Я., Забалуев Н.Н. Топологические алгоритмы трассировки многослойных печатных плат. – М.: Радио и связь, 1989, – 152 с.
139. Пилипака С.Ф. Констрування поверхонь та їх неперервне згинання в кшцеві форми на основі управління натуральними параметрами. Автореф. дисс. доктора техн. наук. Киев, КЛУБА, 2000.: 35 с.
140. Поликарпов Ю.В. Разработка: и исследование специального метода оперативного управления формой кривых и поверхностей. Автореф. дисс.канд. техн. наук. -М.: МАИ; 1979.: 16 с.
141. Полозов В.С., Буденков О.А., Ротков С.И. и др. Автоматизированное проектирование. Геометрические и графические задачи. – М.: Машиностроение, 1983. – 280 с.
142. Полубасов О.Б. Глобальная минимизация количества межслойных переходов // Технология и конструирование в электронной аппаратуре, 2001, № 2, С.3-9.
143. Попов Ю. И. Оптимальное трассирование газосборных сетей на месторождениях: Автореферат диссертации на соискание учёной степени канд. техн. наук.— М.: МИНХ и ГП, 1981.— 22 с.
144. Похвалинский А.Б. Геометрическое моделирование трехмерных объектов методом конструктивной геометрии // Тез. докл. IV Всесоюз. конф. по проблемам машинной графики. - Протвино, 1987. с 108- 126.
145. Похвалинский А.Б. Средства геометрического моделирования трехмерных объектов в САПР. Автореф. дисс. канд. техн. наук. - Киев, КИСИ, 1986.: 24 с.

146. Прохоров А.Ф. Конструктор и ЭВМ. – М.: Машиностроение, 1987. – 272 с.
147. Принс М. Машинная графика и автоматизация проектирования. Пер. с английского. М.: Сов. радио, 1975. - 280 с.
148. Пухов А.А. Автоматизация проектирования дозвуковых грузопассажирских самолетов. Автореф. дисс. на соиск. уч. степ. д-ра техн. наук. – М.: МАИ. – 2005. – 32 с.
149. Радзивиллович В.В. Разработка геометрических моделей криволинейных: поверхностей для гидродинамических расчётов; Автореф. дисс. доктора техн. наук. М., МАДИ, 1990.: 19 с.
150. Рвачев В.Л. Геометрические приложения алгебра логики.- Киев: Техника, 1967. – 211-212 с.
151. Рвачев В.Л. Теория R - функций и некоторые ее приложения. - Киев: Наукова думка, 1982. - 552 с.
152. Реквиша А.А.Г., Воулкер Г.Б. Булевы операции при объемном, моделировании. Алгоритмы определения и соединения границ. // ТИНЭР, 1985, № I. - с.32-49.
153. Рогоза Ю.А. Задание геометрических объектов в дискретном пространстве. МАИ, М.: 1989.-6 е.- Рукопись деп. в ВИНТИ 27.12.89. №7690-В-89.
154. Рогоза Ю.А., Сморгачев Э.К.. Об одной модели в позиционных задачах инженерной графики. Омский политехнический институт. - Омск: 1988. - 6с. - рукопись деп. в ВИНТИ 21.02.89 №1110-В-89.
155. Ротков С.И. Анализ некоторых систем геометрии и графики пространственных объектов. // В сб. Проблемы информационных систем. М., МЦНТИ, 1988, п.5, с. 76
156. Ротков С.И., Шишова Н.А. Теоретико - множественные операции над многогранными объектами сложных структур. // В сб. Автоматизация обработки сложной графической информации. Горький, 1987, с 50

157. Ротков С.И., Шишова Н.А. Теоретико - множественные операции над произвольными многогранниками. // В сб. Материалы 9-ой конференции молодых ученых ГГУ и НИИ Механики. Деп. в ВИНТИ, 15.01.85, п.391.
158. Рыжов Н.Н. О теории каркаса. // Тр. УДН им. Лумумбы, №1 (II) 1963 – с.9-19.
159. Рыжов Н.Н., Каркасная теория задания и конструирования поверхностей. // Тр.УДН им.Лумумбы, 1967, №3 (XXVI), с.3-12.
160. Рыжов Н.Н. Определитель поверхности и его применение. // Тр.УДН им.Лумумбы, том. III, Прикладная геометрия, серия "Математика", вып. 4, 1971. - с.3-17.
161. Сакиева М.К. Геометрическое моделирование конфигурации инженерных сетей: на примере нефте- и газопроводов. Автореф. дис. канд. техн. наук: 05.01.01.— Алма-Аты, 2002.— 24 с.
162. Сакиева М. К., Есмухан Ж. М. Оптимальные связывающие линии // Труды международной конференции «Молодые ученые — 10-летию независимости Казахстана».— Алматы: КазНТУ, 2001,— ч. 1.— С. 450—453.
163. Селютин В.А. Машинное конструирование электронных устройств. – М.: Сов. радио, 1977. – 384 с.
164. Семенов О.И. Автоматизация проектно-конструкторских работ и технологической подготовки производства в машиностроении, т.1.- Минск: Высшая школа, 1976.- 352 с.
165. Семенов О.И. Введение в системы автоматизации проектирования. Минск: Наука и техника, 1979. - 85 с.
166. Семенов О.И. Некоторые итоги создания и задачи развития САПР общемашиностроительного применения // В кн.: II Всесоюзное координационное совещание по автоматизации проектирования в отраслях машиностроения. Минск: ИТК АН БССР, 1981, с. 3-11.

167. Семенков О. И., Васильев В. П. Основы автоматизации проектирования поверхностей с использованием базисных сплайнов, Акад. наук БССР. Ин-т технич. кибернетики . – Минск. : Наука и техника, 1987 . – 167 с.
168. Сидоренко С.М. Вычислительная геометрия в машиностроении. М.: Машиностроение, 1983г. - 160 с.
169. Ситу Лин. Разработка методов и геометрических моделей анализа незаполненных пространств в задачах размещения. Автореф. дисс. на соиск. уч. степ. канд. техн. наук. М.: МАИ, 2011, - 24 с.
170. Ситу Л., Хтун Н. Н., Маркин Л. В. Рецепторные геометрические модели в задачах автоматизированной компоновки технического отсека легкого самолет // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №47, 2011 г.
171. Ситу Лин. Дискретные модели выявления формы незаполненных пространств. // Научно – практическая конференция студентов и молодых учёных МАИ «Инновации в авиации и космонавтике – 2010». Тезисы докладов. – М.: Изд-во МАИ. – 2010. – С.192.
172. Скидан И.А. Геометрическое моделирование кинематических поверхностей в специальных координатах. Автореф. дисс. . доктора техн. наук. –М.: МАДИ, 1989.: 37 с.
173. Смеляков С.В. О построении оптимальной трассы, имеющей минимальное число пересечений с данной сетью // Вычислительная техника и машиностроение.— Минск, 1983.— №3.— С. 19—27.
174. Смеляков С.В., Стоян Ю.Г. Математическая модель некоторых задач оптимизации на путях. – Изв. АН СССР. Сер. Техническая кибернетики, 1981, №4, с.180-188.
175. Смирнов О.Л., Падалко С.Н., Пиявский С.А. САПР: формирование и функционирование проектных модулей. – М.: Машиностроение, 1987. – 272 с.
176. Советов Б. Я. Моделирование систем. – М. : Высшая школа, 2003 – 242 с.

177. Стародетко Б.А. Геометрическое интерполирование и аппроксимация // В кн.: Автоматизация процессов проектирования. Минск: ИТК АН БССР, 1978, вып. 3, с.3-15.
178. Стародетко Б.А. Методы описания и преобразования геометрической информации в автоматизированных системах технической подготовки производства. Автореф. дисс. . доктора техн. наук. –Минск, ИТК АН БССР, 1974, - 42 с.
179. Стародетко Е.А. Математическое моделирование лекальных поверхностей. Минск: Наука и техника, 1984. - 126 с.
180. Стародетко Е.А. Элементы вычислительной геометрии.- Минск, Наука и техника, 1986. – 239 с.
181. Стешенко В.Б. Практика автоматизированного проектирования радиоэлектронных устройств. – М.: Нолидж, 2002 – 768 с.
182. Стоян Ю.Г. Размещение геометрических объектов. - Киев: Наукова думка, 1975. - 239 с.
183. Стоян Ю.Г., Винарский В.Я. Алгебро-топологические свойства объектов. Харьков, 1981. - 34 с. (Препринт ИПМаш АН УССР).
184. Стоян Ю.Г., Гиль Н.И. Методы и алгоритмы размещения плоских геометрических объектов.- Киев: Изд-во Наукова думка, 1976. - 249 с.
185. Стоян Ю.Г., Панасенко А.А. Периодическое размещение геометрических объектов. - Киев: Наукова думка, 1978. - 176 с.
186. Стоян Ю.Г., Кулиш Е.Н. Автоматизация проектирования компоновки оборудования летательных аппаратов. - М.: Машиностроение, 1984. -192с.
187. Стоян Ю.Г., Смеляков С.В., Аристова И.В., Алисейко Е.В. О сведении задачи телесной трассировки к задаче поиска оптимальной манхеттеновой трассы // Киев, Теория и методы автоматизации проектирования, вып. 1., 1984, с. 5-9.
188. Стоян Ю. Г., Яковлев С. В.. Математические модели и оптимизационные методы геометрического проектирования. - АН УССР, Ин-т пробл. машиностроения, Киев, Наук. Думка, 1986. – 265 с.

189. Тищенко М.Н., Артамонов Б.Л., Завалов О.А., Маслов А.Д. Методология применения компьютерных технологий при проектировании винтокрылых летательных аппаратов и их систем // Электронный журнал “Прикладная геометрия”, вып. 11, № 22 (2009) , стр. 105-125.
190. Толок А.В. Применение воксельных моделей в процессе автоматизации математического моделирования // Автоматика и телемеханика. 2009, № 6 . С.167-180.
191. Толок А.В., Толок Н.Б., Губина Е.Н. Распараллеливание процесса рекурсивных вычислений в задаче дихотомического разбиения куба // Прикладная Информатика. 2011. № 4 (34). С. 84-89.
192. Тузов А.Д. Сглаживание функций, заданных таблицами // Вычислительные системы. Новосибирск. - 1976. - вып. 68. -С.61-66.
193. Уваров А.С. Программа P-CAD. Электронное моделирование. –М.: Изд-во “Диалог-МИФИ”, 2008, - 192 с.
194. Флеров Ю.А. Математические методы решения геометрических задач в САПР.// В кн.: II Всесоюзное координационное совещание по автоматизации проектирования в отраслях машиностроения. Минск: ИТК АН БССР, 1981, с.90-97.
195. Формалев В.Ф. Основы построения и математическое обеспечение системы автоматизированного проектирования летательных аппаратов. Автоматизация проектных расчетов. - М.: МАИ, 1980. - 42 с.
196. Фролов С.А. Кибернетика и инженерная графика. -М.: Машиностроение, 1974г. - 200 с.
197. Хасанов В.Х. Разработка и исследование метода геометрического моделирования и расчёта многопараметрических линий и поверхностей. Автореф. дисс. . канд. техн. наук. М., МАДИ, 1982.: 18 с.
198. Хокс Б. Автоматизированное проектирование и производство. Пер. с англ. М.: Мир, 1991. – 296 с.

199. Хорафас Д., Легг С. Конструкторские базы данных. Пер. с англ. М.: Машиностроение, 1990. – 224 с.
200. Хтун Н. Н., Ситу Л., Маркин Л. В. Рецепторные геометрические модели в задачах автоматизированной компоновки технического отсека легкого самолет // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №47, 2011 г. URL: <http://www.mai.ru/science/trudy/published.php>
201. Хтун Н. Н., Тайк Ч., Маркин Л. В. Исследование алгоритмов использования рецепторных геометрических моделей в задачах телесной трассировки авиационной техники // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №69, 2013 г. URL: <http://www.mai.ru/science/trudy/published.php>
202. Хтун Н. Н., Маркин Л. В., Соседко А.А. Применение рецепторных геометрических моделей в задачах автоматизированной компоновки авиационной техники // Труды МАИ, электр. журнал (<http://www.mai.ru>), вып. №72, 2014 г. URL: <http://www.mai.ru/science/trudy/published.php>
203. Шпур Г., Краузе Ф.-Л. Автоматизированное проектирование в машиностроении. Пер. с немецкого. - М.: Машиностроение, 1988. – 648 с.
204. Энгельке У. Как интегрировать САПР и АСТПП. Пер. с англ. - М.: Машиностроение, 1990. – 320 с.
205. Юрьев А.С. Справочник по расчетам гидравлических и вентиляционных систем. –СПб., АНО НПО “Мир и семья”, 2001. -1154 с.
206. Яковлев С. В. Методы и алгоритмы решения оптимизационных задач геометрического проектирования // Автореферат дис. канд. физ. матем. наук.— Киев, 1982. - С. 20.
207. Якунин В.И. Теоретические основы построения интегрированных систем геометрического проектирования сложных поверхностей летательных аппаратов. Автореф. дисс. доктора техн. наук. М., МАИ, 1982.: 41 с.

208. Якунин В.И. Методологические вопросы геометрического конструирования технических объектов // Начертательная геометрия и машинная графика в практике решения инженерных задач. Омск: изд. ОмПИ, 1987. - С.4-7.
209. Якунин В.И. Современные проблемы и перспективы научных исследований в прикладной геометрии // Начертательная геометрия и машинная графика в практике решения инженерных задач. Омск: изд. ОмПИ, 1986. - С.12.
210. Яценко В. А. Геометрическое моделирование в задаче оптимального проектирования инженерных сетей // Автореферат дисс. канд. техн. наук.—Киев, 1988. - С.16.
211. A. Albano, and G. Sapuppo: Optimal Allocation of Two-Dimensional Irregular Shapes Using Heuristic Search Methods, IEEE Transactions on Systems, Man and Cybernetics (1980), 10, pp. 242-248.
212. Atkinson H. H., Gargantini I., Walsh T.R.S.: Filling by Quadrants or Octants, - Computer Vision, Graphics and Image Processing, v.33, №2, Feb, pp. 138-155.
213. Bortfeldt A.: A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces, European Journal of Operational Research 172(2005), pp. 814–837.
214. Brian Hall: Artificial Intelligence for game developers, E-Institute Publishing, Inc. Game Institute, U.S.A, 2002, pp.13-15.
215. J. Cagan, G. Reddy : An Improved Shape Annealing Algorithm for Optimally Directed Shape Generation, Artificial Intelligence in Design (1992), pp. 307-324.
216. Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L.; Stein, Clifford : Section 24.3: Dijkstra's algorithm. Introduction to Algorithms (Second ed.). MIT Press and McGraw–Hill. (2001): pp. 595–601.
217. Delling, D. and Sanders, P. and Schultes, D. and Wagner, D.: Engineering route planning algorithms. Algorithmics of large and complex networks. Springer (2009): pp. 117–139.

218. E. W. Dijkstra : A note on two problems in connexion with graphs, *Numerische Mathematik*. V. 1 (1959), pp. 269-271.
219. Frieder G., Gordon D., Reynolds R. A.: Back-to Front display of Voxel-Based Objects, *IEFE Computer Graphics and Applications*, v. 5, №1, Jan. 1985, pp. 5-60.
220. Gargantini I.: Linear Octrees for Fast Processing of Three - Dimensional Objects, *Computer Graphics and Image Processing*, v. 20, №4, Dec. 1982, pp. 565-574.
221. Gargantini I., Walsh T.R., Wu O.L.: Viewing Trans-Formations of Voxel-Based Objects via linear Octrees, *IEEE Computer Graphics and Applications*, 1986, №10, pp. 12-21.
222. J. A. George: Nested dissection of a regularfinite element mesh, *SIAM J. Numer. Anal.*, 10 (1973), pp. 345-363.
223. P.C. Gilmore; R. E. Gomory: A Linear Programming Approach to the Cutting-Stock Problem. *Operations Research*, Vol. 9, No. 6. (Nov. -Dec., 1961), pp. 849-85.
224. Hart, P. E., Nilsson, N. J., Raphael, B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics* SSC4 4 (2) (1968): pp. 100–107.
225. Jacking C.L., Tanimoto S.L.: Oct-trees and Their Use in Representating Three - Dimensional Objects, *Computer Graphics and Image Processing*, V14 , №3, Nov. 1980, pp. 249-270.
226. Masatomo Kanehara, Satoshi Kagami, James.J Kuffner, Simon Thompson, Hiroshi Mizoguchi: Path shortening and smoothing of grid-based path planning with consideration of obstacles, Tokyo, 2000, pp.991-996.
227. Knuth, D.E: A Generalization of Dijkstra's Algorithm. *Information Processing Letters* 6 (1) (1977), pp.1–5.
228. Li Guiqing, Li Xianmin, Li Hua: 3D Discrete Clothoid Splinesh, *CGIf01*, Japan, pp. 321-324.

229. Khammapun Khantanapoka and Krisana Chinnasarn: Pathfinding of 2D & 3D Game Real-Time Strategy with Depth Direction A*Algorithm for Multi-Layer, 2009 Eighth International Symposium on Natural Language Processing, Thailand, pp. 184-188.
230. Andrew Lim and Lei Wang, Wenbin Zhu, Hu Qin* : A two-stage tabu search algorithm with enhanced packing heuristics for the 3L-CVRP and M3L-CVRP , Computers & Operations Research, Volume 39, Issue 9, September 2012, pp. 2178 - 2195.
231. A. Lodi, S. Martello, D. Vigo: Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems, INFORMS Journal on Computing 11 (1999), pp. 345–357.
232. Koji Makanae: An application of Parametric Curves to Highway Alignment, Journal of Civil Engineering Information Processing System in 2000, pp. 169-176.
233. S. Martello, P. Toth: Knapsack problems: algorithms and computer implementations, J. Wiley & Sons, Dec 14, 1990.
234. Matthews, James: Basic A* Pathfinding Made Simple, AI Game Programming Wisdom, Charles River Media, 2002.
235. Masoud Nosrati, Ronak Karimi, Hojat Allah Hasanvand.: Investigation of the * (Star) Search Algorithms: Characteristics, Methods and Approaches, World Applied Programming, Vol (2), No (4), April 2012. ISSN: 2222-2510, pp. 251-256.
236. Meagher. D.: Geometric Modeling Using Octree Encoding, Computer Graphics and Image Processing, V.19, №2, June 1982, pp. 129-147.
237. Nyi Nyi Htun. Finding the shortest smooth path in variable size using improved A* algorithm on grid-based receptor model // (<http://www.ucsy.edu.mm/ucsy/635558k.do>) « 11th International Conference On Computer Applications – 2013, Yangon, Myanmar ». –UCSY. – 2013, pp.255-260.
238. Pearl J.: Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley, ISBN 0-201-05594-5 (1984), p. 48.

239. Pearl Judea; Jin H. Kim: Studies in semi-admissible heuristics. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 4 (4)(1982): pp. 392–399.
240. Pearl Judea, Dechter Rina: Generalized best-first search strategies and the optimality of A*, Journal of the ACM 32 (3)(1985): pp. 505–536.
241. Pisinger. D.,Vigo. D., Martello S.: The three-dimensional bin packing problem. Operations Research 48(2000), pp. 256–267.
242. Pohl, Ira: First results on the effect of error in heuristic search, Machine Intelligence 5: (1970): pp. 219–236.
243. Pohl, Ira: The avoidance of (relative) catastrophe, heuristic competence, genuine dynamic weighting and computational issues in heuristic problem solving, Proceedings of the Third International Joint Conference on Artificial Intelligence (IJ-CAI-73) 3. California, USA (August 1973): pp. 11–17.
244. Przemysław Kubiak (Eds.), Mirosław Kutylowski and Jacek Cichoń: Algorithmic Aspects of Wireless Sensor Networks, Third International Workshop, ALGOSENSORS 2007, Wrocław, Poland, July 14, 2007, Revised Selected Papers, volume 4837 of Lecture Notes in Computer Science, Springer, Berlin, 2008, ISBN 978-3-540-77870-7, pp. 99–113.
245. Requcha A.A.G., Voelker H.B., Solid Modeling : A historical summary and contemporary assessment, - IEEE, Computer Graphics and Applications , 1982, v.2, №2, pp. 9-24.
246. Requcha A.A.G.: Representation for Rigid Solid. Theory Methods and Systems. – Computer Surveys, 1980, v.12, №4, pp. 437-464.
247. Robinson, D.F., Foulds, L.R.: Comparison of phylogenetic trees, Mathematical Biosciences 53 (1981), pp. 131–147.
248. Russel, Stuart., Norvig, Peter.: Artificial Intelligence A Modern Approach, (2nd ed.), Upper Saddle River, New Jersey: Prentice Hall, ISBN 0-13-790395-2 (1995). pp. 94 -95 (note 3).
249. Saaty T.L.: The Analytic Hierarchy Process, NY: McGraw Hill, 1980, p. 278.

250. Samet H., Robert E., Webber E.: Hierarhical Data Structures And Algorithms For Computer Graphics, Computer Graphics and Applications, May 1988, pp. 48-68.
251. Stoyan Yu. G., Smelyakov S.V.: An approach to the problems of routing optimization in the regions of intricate shape. – U.S.S.R, 1981, pp. 39-43.
252. Szykman S., J. Cagan : Synthesis of Optimal Non-Orthogonal Routes, ASME Journal of Mechanical Design ,118(3)(1996): pp. 419-424.
253. Simon Thompson, Satoshi Kagami: Smooth Trajectory Planning with Obstacle Avoidance for Car-Like Mobile Robots , The 23th Annual Conference of the Robotics Society of Japan, pp. 498-502.
254. Vigo D., M. Dell'Amico, S. Martello: Heuristic algorithms for single processor scheduling with earliness and flow-time penalties. I. H. Osman, J. P. Kelly (eds.), Meta-Heuristics: Theory and Applications, Kluwer Academic Publishers, Boston, MA (1996), pp. 167–182.
255. D. R. Wichmann: Automated Route Finding on Digital Terrains, COMPSCI 780 Project Report, Graphics Group, Dept. of Computer Science, University of Auckland, New Zealand, February 2004, pp. 107-113.

Приложение 1

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)



«УТВЕРЖДАЮ»

Проректор по учебной работе

М.Ю.Куприков

«30» декабря 2013 г.

АКТ ВНЕДРЕНИЯ

Настоящим актом подтверждается внедрение результатов диссертационного исследования аспиранта кафедры 904 МАИ Ньи Ньи Хтуна в области рецепторных методов геометрического моделирования в учебный процесс кафедры 904 в читаемых кафедрой курсах: **«Методы геометрического моделирования и автоматизации жизненного цикла изделий в CAD/CAM/CAE»** для слушателей факультета повышения преподавателей МАИ (ФПКП МАИ) и **«Методы геометрического моделирования и автоматизированного воспроизведения изделий в CAD/CAM/CAE»** (курс для аспирантов МАИ).

Зам заведующего кафедрой 904
к.т.н. доцент

A handwritten signature in blue ink, consisting of stylized letters.

А.В.Рипецкий

Приложение 2

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ «АСА»

115573, г. Москва, ул. Мусы Джалиля 10/1-131
Телефон: +7-901-530-3439; Факс +7-901-534-1511

А К Т В Н Е Д Р Е Н И Я

Результаты диссертационной работы аспиранта кафедры 904 МАИ Ньи Ньи Хтуна в тестовом режиме внедрены на малом авиастроительном предприятии «АСА» в виде программного комплекса, позволяющего производить трассировку трубопроводов как постоянного, так и переменного сечений в моторном отсеке с учетом заданного минимального радиуса кривизны трубопровода. Программный комплекс прошел тестирование на легком самолете «АСА-2» и позволил выявить возможность прокладки дополнительных воздухопроводов в техотсеке этого самолета и визуализировать результаты проектирования в системе SolidWorks. Полученные результаты позволяют определять возможность дополнительного размещения трубопроводов сложной геометрической формы и изменения существующих при модификации конструкции или последующий модернизации самолета.

Генеральный директор
ООО «АСА»
12 октября 2013 г.

А.А Соседко

