

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М. В. ЛОМОНОСОВА

На правах рукописи
УДК 519.71

Снегова Елена Александровна

СЛОЖНОСТЬ ЗАДАЧИ
О ПРЕДОТВРАЩЕНИИ СТОЛКНОВЕНИЙ

01.01.09 — дискретная математика и математическая кибернетика

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата физико-математических наук

МОСКВА — 2012

Работа выполнена на кафедре Математической теории интеллектуальных систем Механико-математического факультета Московского государственного университета имени М.В. Ломоносова.

Научный руководитель: доктор физико-математических наук,
профессор Гасанов Эльяр Эльдарович

Официальные оппоненты: Михалев Александр Александрович
доктор физико-математических наук,
профессор (ФГБОУ ВПО
«Московский государственный университет
им. М. В. Ломоносова»)

Алексиадис Никос Филиппович
кандидат физико-математических наук,
доцент (ГОУ ВПО «Московский
энергетический институт (технический
университет)»)

Ведущая организация: ГОУ ВПО «Московский государственный
университет приборостроения и
информатики»

Защита диссертации состоится 16 ноября 2012 г. в 16 ч. 45 м. на заседании диссертационного совета Д 501.001.84 при Московском государственном университете имени М.В.Ломоносова по адресу: Российская Федерация, 119991, Москва, ГСП-1, Ленинские горы, д.1, Московский государственный университет имени М.В. Ломоносова, Механико-математический факультет, аудитория 14-08.

С диссертацией можно ознакомиться в библиотеке Механико-математического факультета МГУ имени М.В. Ломоносова (Главное здание, 14 этаж).

Автореферат разослан 16 октября 2012 г.

Ученый секретарь диссертационного
совета Д.501.001.84 при МГУ
доктор физико-математических наук,
профессор

А.О.Иванов

Общая характеристика работы

Актуальность темы. Диссертация посвящена исследованию задач поиска в базах данных движущихся объектов и относится к одному из важнейших разделов теории интеллектуальных систем и математической кибернетики — теории хранения и поиска информации. Рассматриваемая в работе задача о предотвращении столкновений сводится к традиционно трудным динамическим задачам поиска и является актуальной для теории баз данных как с теоретической, так и практической точек зрения, поскольку может быть использована диспетчерскими службами аэропортов, системами управления движением транспорта и в задачах обсчета физических экспериментов на столкновение частиц.

В данной работе рассматривается следующая вариация задачи о предотвращении столкновений. Рассматриваются два потока объектов на плоскости. Один из них называется потоком запросов, а другой — потоком объектов-данных. Движение объектов рассматривается внутри фиксированного квадрата, и каждый объект, появившись на границе квадрата, сообщает системе (внешнему наблюдателю) свои координаты появления и закон движения (будущего движения внутри квадрата), а система фиксирует момент появления объекта. В задаче требуется для каждого запроса перечислить все объекты, которые столкнутся с запросом в процессе движения. В работе также рассматриваются и более специальные случаи данной задачи — когда все объекты-данные имеют одинаковые или похожие законы движения, и, аналогично, запросы имеют одинаковые или похожие законы движения.

Для оценки работы алгоритмов решения задачи о предотвращении столкновений используются четыре основных меры сложности — объем памяти, необходимый для реализации алгоритма, сложность поиска в базе данных объектов, с которыми может произойти столкновение, сложность вставки в базу данных нового объекта и сложность удаления из базы данных имеющегося объекта.

Простейшим способом решения задача о предотвращении столкновений является алгоритм перебора. В этом случае база данных в каждый момент времени будет представлять из себя список объектов-данных, находящихся в этот момент времени внутри квадрата. Для того, чтобы получить ответ на запрос, требуется просмотреть весь список объектов в базе данных, и для каждого объекта проверить, будет ли он находиться с запросом в опасной близости в какой либо момент. Очевидно, что данную операцию можно проделать за время порядка n , где n есть число объектов в базе данных.

Поскольку задача о предотвращении столкновений динамическая, то

необходимо производить обновление базы данных – процедуры вставки и удаления. В алгоритме перебора процедура вставки будет занимать константное время (добавление в список), а процедура удаления будет происходить в момент, когда объект достигает границы квадрата путем поиска в списке объекта, который требуется удалить, а затем процедуры его удаления. Таким образом, в худшем случае сложность удаления будет иметь порядок n .

Так как алгоритм перебора имеет большую сложность поиска, то он является неэффективным. Более эффективными будем считать алгоритмы, имеющие меньшую по порядку сложность поиска без перечисления ответа за счет увеличения объема памяти.

Такие оценки могут быть достигнуты, если законы движения объектов и запросов предопределены, и удастся свести многомерную задачу о предотвращении столкновений к одномерным задачам. В данной работе рассматриваются два вида одномерных задач и приведены все случаи, когда задача о предотвращении столкновений может быть сведена хотя бы к одной из этих двух задач (то есть представлены критерии сводимости).

Другая возможность решить задачу о предотвращении столкновений эффективным способом – это сузить множество объектов, подлежащее перебору. В этом случае удастся получить среднюю сложность поиска без перечисления ответа, равную по порядку квадратному корню от мощности множества объектов.

Задача о предотвращении столкновений похожа по своей формулировке и методам решения на задачу из области вычислительной геометрии о поиске объекта, ближайшего к запросу, которая состоит в следующем: дано множество S из n объектов, для каждого запроса q требуется найти объект s из S , ближайший к q .

В большей части методов решения задачи поиска объекта, ближайшего к запросу, используются структуры данных, основанные на конструкции R -дерева¹ или его модификации – приоритетном R -дереве².

Задача о поиске объекта, ближайшего к запросу, имеет 4 вариации:

1. Случай статических (неподвижных) объектов и статических (неподвижных) запросов: Preparata и Shamos³, Roussopoulos⁴, Cheung и

¹Guttman A. R-Trees: A Dynamic Index Structure for Spatial Searching. In Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data, pp. 47–57, 1984.

²L. Arge, M. de Berg, H. J. Haverkort, and K. Yi. The Priority R-Tree: A Practically Efficient and Worst-Case-Optimal R-Tree. Symp. of the ACM Special Interest Group on Management of Data (SIGMOD), Paris, 2004, pages 347–358.

³Preparata F.P., Shamos M.I.: Computational geometry: An introduction (texts and monographs in computer science), 5th edn. Springer, Berlin, Heidelberg, New York (1993)

⁴Roussopoulos N., Kelley S., and Vincent F. Nearest neighbor queries. In Proceedings of ACM SIGMOD

Fu⁵, Korn и др.⁶

2. Случай статических (неподвижных) запросов и динамических (движущихся) объектов: Berchtold и Ertl⁷, Kollios и Gunopulos⁸.

3. Случай динамических (движущихся) запросов и статических (неподвижных) объектов: Song и Roussopoulos⁹.

4. Случай динамических (движущихся) запросов и динамических (движущихся) объектов наиболее близок к рассматриваемой в данной работе задаче о предотвращении столкновений. Большинство алгоритмов для этого случая основываются на повторных применениях алгоритмов для решения обычной задачи о нахождении ближайшего соседа, что приносит значительные издержки¹⁰. Тао и др.¹¹ предложили алгоритм в котором ответ на запрос формируется из трех компонент, а база данных постоянно обновляется. Случай предопределенного движения объектов рассмотрен Attalah'ом¹².

Цель работы. В работе ставится цель как можно более полно описать все случаи задачи о предотвращении столкновений, допускающие эффективные решения, а также привести конкретные алгоритмы решения этих задач. При этом, алгоритм считается эффективным, если он имеет сложность (поиска без перечисления ответа, вставки и удаления) меньше по порядку в среднем или худшем случае, чем сложность поиска в алгоритме перебора.

Важным направлением исследования является выявление одномерных геометрических задач поиска, к которым могут быть сведены некоторые случаи задачи о предотвращении столкновений, а также формулировка и доказательство критериев сводимости задачи о предотвращении столкновений к данным одномерным задачам.

Для наиболее распространенных законов движения объектов,

International Conference on Management of Data, San Jose, USA, 1995.

⁵Cheung K. L. and Fu A. W. Enhanced Nearest Neighbour Search on the R-tree. SIGMOD Record, 27(3): 16–21, 1998.

⁶Korn F., Sidiropoulos N., Faloutsos C., Siegel E., and Protopapas Z. Fast nearest neighbor search in medical image databases. In Proceedings of International Conference on Very Large Database, Mumbai, India, 1996.

⁷Berchtold, S., Ertl, B., Keim, D.A., Kriegel, H.P., Seidl, T.: Fast nearest neighbor search in high-dimensional space. In: Proceedings of the International Conference on Data Engineering, pp. 209–218 (1998)

⁸Kollios, G., Gunopulos, D., Tsotras, V.J.: Nearest neighbor queries in a mobile environment. In: Proceedings of the International Workshop on Spatio-Temporal Database Management, pp. 119–134 (1999)

⁹Song Z. and Roussopoulos N. K-Nearest Neighbor Search for Moving Query Point. In Proceedings of the 7th International Symposium on Spatial and Temporal Databases, pp. 79–96, 2001.

¹⁰Seidl T. and Kriegel H. Optimal multi-step k-nearest neighbor search. In Proceedings of ACM SIGMOD International Conference on Management of Data, Seattle, USA, 1998.

¹¹Tao Y., Papadias D., Shen Q. Continuous Nearest Neighbor Search. In VLDB, 2002.

¹²Atallah M. Dynamic Computational Geometry. Proc. 24th Annu. IEEE Sympos. Found. Comput. Sci, 1983.

являющихся аналитическими функциями, ставится цель получить критерии сводимости к одномерным задачам поиска в явном виде, так, чтобы по одному виду закона движения можно было понять, имеется ли сводимость и тем самым эффективное решение.

Также исследуются случаи задачи о предотвращении столкновений, которые не допускают решения путем сведения к одномерным задачам, но "накрываются" одной или несколькими простыми геометрическими задачами поиска, что позволяет сократить перебор.

Методы исследования. В работе используются методы дискретной математики, в частности комбинаторики, теории графов и теории управляющих систем, а также методы теории вероятностей и математического анализа.

Научная новизна. Работа нацелена на исследование возможности эффективного решения задачи о поиске движущихся объектов. Наиболее эффективные существующие ранее алгоритмы решения задач, близких к рассматриваемой в данной работе задаче, имели сложность в худшем случае, равную по порядку квадратному корню от количества объектов в базе данных.

В работе впервые были предложены алгоритмы решения данной задачи, имеющие логарифмическую сложность поиска/вставки/удаления объекта.

В работе получены следующие результаты:

1. Получены критерии сводимости задачи о предотвращении столкновения к задаче одномерного интервального поиска. На основе этой сводимости был предложен алгоритм решения задачи о предотвращении столкновения, имеющий логарифмическую сложность поиска/вставки/удаления с линейными затратами памяти.

2. Получен критерий сводимости задачи о предотвращении столкновений к задаче о прокалывании.

3. Для случая законов движения, являющихся аналитическими функциями, получен критерий сводимости к одномерным задачам поиска.

4. Разработаны приближенные алгоритмы решения задачи о предотвращении столкновений методом сведения к нескольким одномерным задачам поиска.

5. Для общего случая задачи о предотвращении столкновений разработан алгоритм, имеющий среднее время поиска, вставки и удаления порядка \sqrt{n} , где n – размер базы данных.

Практическая ценность. Результаты работы могут быть использованы в реальных динамических системах управления движением.

Примером ситуации, к которой может быть применена задача о предотвращении столкновений, является слежение за столкновениями двух

перпендикулярных потоков частиц, движение двух потоков самолетов над аэропортом в выделенной плоскости или движение потоков автомобилей на перекрестке.

Логарифмическое время обработки данных в предлагаемых алгоритмах, позволяет использовать их для очень больших баз данных и при жестких временных ограничениях.

Апробация работы. Результаты работы докладывались на Международной конференции "Современные проблемы математики, механики и их приложений посвященной 70-летию ректора МГУ академика В.А.Садовниченко (30 марта - 2 апреля 2009 г., Москва); XVI и XVII конференциях студентов, аспирантов и молодых ученых "Ломоносов-2009" и "Ломоносов-2010 секция "Математика и механика" (Москва, 2009, 2010); X Международном семинаре "Дискретная математика и ее приложения" (Москва, 1-5 февраля 2010 г.); XIV восточно-европейской конференции по достижениям в базах данных и информационных системах "Fourteenth East-European Conference on Advances in Databases and Information Systems" (Нови-Сад, Сербия, 2010); X Международной конференции "Интеллектуальные системы и компьютерные науки" (Москва, 5-10 декабря 2011 года), а также на семинарах механико-математического факультета МГУ им. М.В.Ломоносова: на семинаре "Теория автоматов" под руководством академика, проф., д.ф-м.н. В.Б. Кудрявцева (2010-2011 г.), неоднократно на семинаре "Вопросы сложности алгоритмов поиска" под руководством проф., д.ф-м.н. Э.Э.Гасанова (2006-2011 гг.), на семинаре "Кибернетика и информатика" под руководством академика, проф., д.ф-м.н. В.Б. Кудрявцева (2011 гг.).

Публикации. По теме диссертации опубликовано 12 печатных работ [1-12].

Структура и объем работы. Диссертация состоит из введения и 5 глав. Объем диссертации 105 стр. Список литературы содержит 30 наименований.

Постановка задачи. Сначала дадим неформальную постановку задачи. Внутри квадрата $[-\rho, 1 + \rho] \times [-\rho, 1 + \rho]$ прямолинейно движутся точечные объекты, причем запросы появляются на левой стороне прямоугольника $[-\rho, 1 + \rho] \times [0, 1]$, а объекты – на нижней стороне прямоугольника $[0, 1] \times [-\rho, 1 + \rho]$.

Объекты, появляющиеся на левой стороне квадрата, будем называть *объектами-данными* или просто *объектами* (они пронумерованы в порядке своего появления), а на нижней стороне квадрата – *объектами-запросами* или просто *запросами*.

В момент появления каждый объект и каждый запрос сообщают

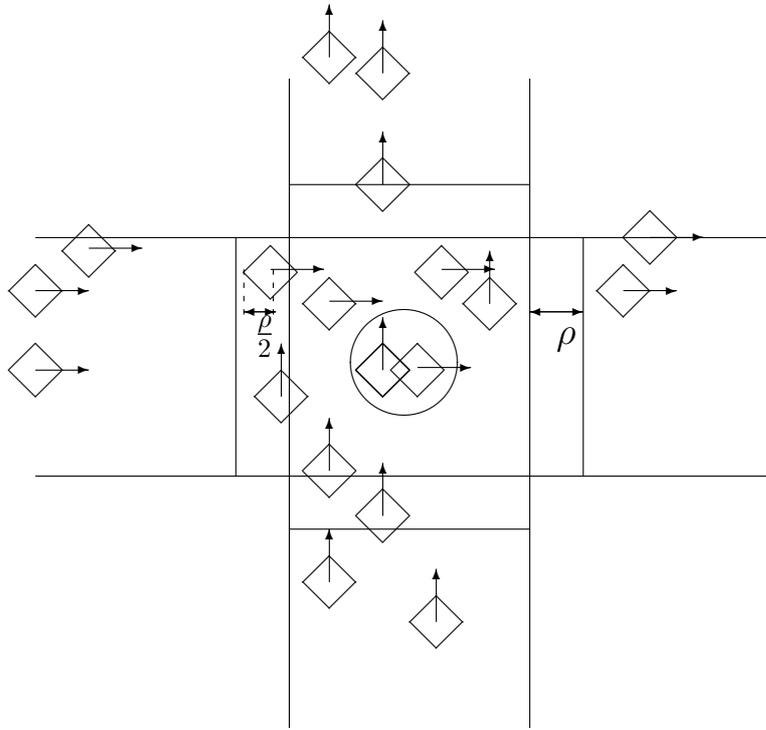


Рис. 1: Движение объектов и запросов в задаче о предотвращении столкновений. Кругом отмечен пример столкновения объекта и запроса.

системе свои координаты и закон движения, а система фиксирует время его появления.

В задаче требуется для каждого запроса перечислить все объекты, которые появились до него и с которыми он, в течение своего движения внутри прямоугольника $[-\rho, 1 + \rho] \times [0, 1]$, будет находиться в состоянии опасной близости, то есть на расстоянии по Манхэттену меньшем, чем ρ .

Оси координат расположим вдоль сторон квадрата, таким образом, что у объектов координата по оси x в начальный момент равна $-\rho$ и вектор скорости параллелен оси x , а у запросов координата по оси y в начальный момент равна $-\rho$, а вектор скорости параллелен оси y .

По-другому, можно рассматривать объекты и запросы как окружности по Манхэттену радиуса $\frac{\rho}{2}$. В этом случае в задаче требуется для каждого запроса перечислить все объекты, которые появились до него и с которыми он столкнется в процессе движения.

Схематично движение объектов и запросов в задаче о предотвращении столкновений изображено на рисунке 1.

Теперь формализуем постановку задачи. Опишем задачу о предотвращении столкновений при фиксированных законах движения объектов и запросов, имеющую несколько параметров:

- 1) параметр $\rho \in (0, \frac{1}{2})$, обозначающий *расстояние опасной близости* по Манхэттену;
- 2) дифференцируемая строго возрастающая функция $f : \mathbb{R}^+ \rightarrow [-\rho, 1 +$

$\rho]$, такая что $f(0) = -\rho$, называемая *законом движения объектов*;

3) дифференцируемая строго возрастающая функция $f_q : \mathbb{R}^+ \rightarrow [-\rho, 1 + \rho]$, такая что $f_q(0) = -\rho$, называемая *законом движения запроса*;

4) бесконечная последовательность $\mathbb{O} = \{o_1, o_2, \dots, o_n, \dots\}$, такая что $o_i = (t_i, y_i)$, $y_i \in [0, 1]$, а t_i – действительные числа, образующие возрастающую последовательность $t_0 < t_1 < t_2 < \dots < t_n \dots$, называемая *потокком объектов*.

Четверку $(f, f_q, \rho, \mathbb{O})$ назовем *задачей о предотвращении столкновений*.

Введем еще два параметра, которые понадобятся в дальнейшем: параметр $\tau_{max} = f^{-1}(1 + \rho)$, называемый временем движения объектов и параметр $\tau_{max}^q = f_q^{-1}(1 + \rho)$, называемый временем движения запроса.

Предполагаем, что объект $o_i = (t_i, y_i)$ появляется на границе прямоугольника $[-\rho, 1 + \rho] \times [0, 1]$ в момент времени t_i в точке с координатами $(-\rho, y_i)$ и движется по закону движения f параллельно оси x , то есть в момент $t \in [t_i, t_i + \tau_{max}]$ объект o_i находится внутри прямоугольника $[-\rho, 1 + \rho] \times [0, 1]$ в точке с координатами $(f(t - t_i), y_i)$.

Запросом q назовем пару (t_q, x) , где $t_q \in \mathbb{R}$, а $x \in [0, 1]$.

Предполагаем, что запрос $q = (t_q, x)$ появляется на границе прямоугольника $[0, 1] \times [-\rho, 1 + \rho]$ в момент времени t_q в точке с координатами $(x, -\rho)$ и движется по закону движения f_q параллельно оси y , то есть в момент $t \in [t_q, t_q + \tau_{max}^q]$ запрос q находится внутри прямоугольника $[0, 1] \times [-\rho, 1 + \rho]$ в точке с координатами $(x, f_q(t - t_q))$. Скажем, что объект $o_i = (t_i, y_i)$ и запрос $q = (t_q, x)$ *сталкиваются*, если существует момент времени $t \in \mathbb{R}$, такой что $|f(t - t_i) - x| + |y_i - f_q(t - t_q)| \leq \rho$.

В задаче требуется для произвольного запроса перечислить все объекты из потока \mathbb{O} , которые появились до него и с которыми он столкнется в процессе своего движения, то есть *ответом на запрос* $q = (t_q, x)$ при расстоянии опасной близости ρ является множество

$$J(f, f_q, \rho, \mathbb{O}, q) = \{o_i = (t_i, y_i) \in \mathbb{O} \mid t_i < t_q \text{ и } \exists t : |f(t - t_i) - x| + |y_i - f_q(t - t_q)| \leq \rho\}.$$

Библиотекой $V(t)$ в момент времени t назовем множество объектов из потока \mathbb{O} , находящихся внутри прямоугольника $[-\rho, 1 + \rho] \times [0, 1]$ в момент времени t , т.е. $V(t) = \{o_i = (t_i, y_i) \in \mathbb{O} : t_i \leq t \leq t_i + \tau_{max}\}$.

Введем последовательность, объединяющую моменты появления и исчезновения объектов $\{s_i\}_{i=1}^{\infty} = \{t_i\}_{i=1}^{\infty} \cup \{t_i + \tau_{max}\}_{i=1}^{\infty}$ и заметим, что по сути $V(t)$ – это кусочно-постоянная функция $V : \mathbb{R} \rightarrow 2^{\mathbb{O}}$, меняющаяся в моменты появления и исчезновения объектов s_i , $i \in \mathbb{N}$.

Решать задачу $(f, f_q, \rho, \mathbb{O})$ в приведенной выше постановке не представляется возможным, поскольку множество \mathbb{O} бесконечное и невозможно его целиком хранить в базе данных.

Допустим, что в каждый момент времени t имеются данные не про все множество \mathbb{O} , а про его "текущее" подмножество $V(t)$.

Введем множество ответа на запрос $q = (t_q, x)$ при библиотеке $V(t)$ в момент времени t :

$$J(f, f_q, \rho, V(t), q) = \\ = \{o_i = (t_i, y_i) \in V(t) \mid \exists t' : |f(t' - t_i) - x| + |y_i - f_q(t' - t_q)| \leq \rho\}$$

и заметим, что при $t_q \in [\max_{s_i \leq t} s_i, \min_{s_i > t} s_i]$ $J(f, f_q, \rho, \mathbb{O}, q) = J(f, f_q, \rho, V(t), q)$.

Таким образом, в каждый момент времени t имеется информация о библиотеке $V(t)$ и содержательно задача о предотвращении столкновений состоит из двух компонент:

1. Обновление библиотеки $V(t)$ в моменты времени $s_i, i \in \mathbb{N}$. В каждый момент $t_i, i \in \mathbb{N}$, происходит добавление объекта $o_i = (t_i, y_i)$ в "текущую" библиотеку, а в каждый момент $t_i + \tau_{max}, i \in \mathbb{N}$, происходит удаление объекта $o_i = (t_i, y_i)$ из "текущей" библиотеки.

2. Поиск ответа на запрос $q = (t_q, x)$ в библиотеке $V(t)$, при условии, что $t_q \in [\max_{s_i \leq t} s_i, \min_{s_i > t} s_i]$ ¹³.

Задачу о предотвращении столкновений в момент времени t будем обозначать как четверку $(f, f_q, \rho, V(t))$.

Алгоритм решения задачи о предотвращении столкновений.

Мы будем рассматривать задачу о предотвращении столкновений как динамическую задачу поиска. Обычно предполагается, что алгоритмы решения динамических задач поиска содержат в своей памяти базу данных, представленную некоторой структурой. Как правило структуры данных описываются на языке графов, например, на языке информационных графов¹⁴. При этом алгоритм решения динамических задач поиска воспринимается как совокупность трех процедур, работающих над этой структурой данных: процедуры поиска, процедуры вставки и процедуры удаления. Процедура поиска основная и позволяет решать главную задачу — выбирать в базе данных объекты, удовлетворяющие запросу. Процедуры вставки и удаления служат для вставки в базу данных новых объектов и удаления из базы имеющихся объектов, каждая из этих процедур изменяет

¹³Заметим, что t также принадлежит отрезку $[\max_{s_i \leq t} s_i, \min_{s_i > t} s_i]$, то есть в библиотеку $V(t)$ могут поступать запросы с моментом появления равным t или близким к t

¹⁴Гасанов Э. Э., Кудрявцев В.Б. Теория хранения и поиска информации. — М.: ФИЗМАТЛИТ, 2002.

структуру данных, чтобы она позволяла корректно решать задачу поиска. При этом структуру данных выбирают таким образом, чтобы каждая из этих процедур была как можно более эффективной. Чтобы строго определить процедуры поиска, вставки и удаления вводят множество элементарных операций, каждая из которых имеет свою сложность. Сами процедуры описываются либо на естественном языке, либо на специальном (алгоритмическом, языке блок-схем, языке информационных графов) таким образом, что для каждого фиксированного входного блока данных (запроса для процедуры поиска, или объекта данных для процедур вставки и удаления) процедура сводится к однозначной последовательности элементарных операций, сумма сложностей которых и определяют сложность данной процедуры на данном входном блоке.

В нашем случае при решении задачи о предотвращении столкновений структуры данных мы будем представлять в виде нагруженных графов. Множество элементарных операций будет содержать все арифметические операции над вещественными числами и все операции сравнения вещественных чисел. Также будем считать, что в множество элементарных операций содержит операции вычисления значений функций законов движения в любой точке, а также операции вычисления значений производных от законов движения, и обратных функций от законов движения и их производных. При этом будем считать, что каждая операция имеет фиксированную сложность, причем за единицу измерения примем сложность операций сравнения, считая их самыми простыми. Чтобы иметь возможность преобразовывать графы, будем считать, что множество элементарных операций содержит операции вставки и удаления вершин и ребер графа, операции изменения и вычисления нагрузок вершин и ребер, а также некоторые более сложные составные операции, каждая из которых выполняет некоторое локальное преобразование графа и имеет константную сложность, где опять же в качестве единицы измерения выступают операции сравнения. Примерами таких операций могут служить операции склейки и переброски вершин. Если неформально описывать эти операции, то операция *переброски* может быть применена к внутренней вершине f графа, имеющей четырех сыновей. Для операции переброски образуем новую вершину g . Два левых сына оставим сыновьями f , а два правых переделаем в сыновей g . Затем сделаем g братом f , сделав его сыном отца f . Операция *склейки* может быть применена к двум вершинам, имеющим общего родителя. Она заключается в простом объединении (склеивании) этих двух вершин.

Пусть $\mathbb{O} = \{o_1 = (t_1, y_1), o_2 = (t_2, y_2), \dots, o_i = (t_i, y_i), \dots\}$ — произвольный поток объектов и $\{s_i\}_{i=1}^{\infty} = \{t_i\}_{i=1}^{\infty} \cup \{t_i + \tau_{max}\}_{i=1}^{\infty}$ —

последовательность, объединяющая моменты появления и исчезновения объектов. Алгоритмом решения задачи о предотвращении столкновений $(f, f_q, \rho, \mathbb{O})$ будем называть совокупность трех процедур поиска, вставки и удаления, таких, что для любого $i \in \mathbb{N}$ если вызывать процедуру вставки во все моменты $t_j \leq s_i$, а процедуру удаления во все моменты $t_j + \tau_{max} \leq s_i$ для объектов o_j , то для любого запроса $q = (t_q, x)$ такого, что $s_i \leq t_q < s_{i+1}$, процедура поиска в ответ будет выдавать множество $J(f, f_q, \rho, V(t_q), q)$.

Пусть A – алгоритм, решающий задачу о предотвращении столкновений. Тогда сложность поиска по алгоритму A на запросе q при библиотеке $V(t)$ есть величина, равная сумме сложностей всех элементарных операций, выполненных процедурой поиска при подаче на ее вход запроса q при условии, что на момент поиска структура данных соответствовала библиотеке $V(t)$. Сложность поиска всегда включает в себя сложность перечисления ответа, которая как минимум равна длине ответа на запрос. Поэтому в перечислительных задачах поиска как правило исследуют сложность поиска без перечисления ответа. Сложностью поиска без перечисления ответа на запросе q назовем величину, равную сложности на запросе q минус мощность ответа на запрос q . Обозначим эту сложность $T_A(f, f_q, \rho, q, V(t))$.

Сложность вставки по алгоритму A объекта o в библиотеку $V(t)$ обозначим $S_A(f, f_q, \rho, o, V(t))$ и будем считать равной сумме сложностей всех элементарных операций, выполненных процедурой вставки при подаче на ее вход объекта o при условии, что на момент вставки структура данных соответствовала библиотеке $V(t)$.

Сложность удаления по алгоритму A объекта o из библиотеки $V(t)$ обозначим $R_A(f, f_q, \rho, o, V(t))$ и будем считать равной сумме сложностей всех элементарных операций, выполненных процедурой удаления при подаче на ее вход объекта o при условии, что на момент удаления структура данных соответствовала библиотеке $V(t)$.

Объемом алгоритма A для библиотеки $V(t)$ назовем число ребер в графе, описывающем структуре данных, соответствующей библиотеке $V(t)$, и обозначим $Q_A(f, f_q, \rho, V(t))$.

Содержание работы

Работа состоит из введения и пяти глав. Введение содержит обзор литературы, постановку задачи и формулировку основных результатов.

В первой главе приводится критерий сводимости задачи о предотвращении столкновений к одномерной задаче интервального

поиска и алгоритм решения задачи в этом случае.

Задачей одномерного интервального поиска назовем пару (I, Z) , где библиотека Z — конечное подмножество множества действительных чисел \mathbb{R} , а множество запросов I — есть множество всех интервалов с концами из \mathbb{R} . Содержательно эта задача состоит в том, чтобы для произвольного запроса $p \in I$ перечислить все те и только те точки из Z , которые попадают в интервал p .

Ответ на запрос $p \in I$ при библиотеке Z в задаче одномерного интервального поиска есть множество $J(p, Z) = \{z \in Z : z \in p\}$.

Будем говорить, что задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче одномерного интервального поиска, если существуют такие отображения $\varphi, \varphi_1, \varphi_2 : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}$, что для любой библиотеки $V(t)$, любого запроса $q = (t_q, x)$ и любого объекта $o \in V(t)$ верно $o \in J(f, f_q, \rho, V(t), q) \Leftrightarrow \varphi(o) \in J([\varphi_1(q), \varphi_2(q)], Z)$, где $Z = \{\varphi(o) : o \in V(t)\}$.

Обозначим:

$$\begin{aligned} F_L(x, y) &= \min_{\xi \in [0, \rho]} (f_q^{-1}(y + \xi - \rho) - f^{-1}(x + \xi)), \\ F_R(x, y) &= \max_{\xi \in [-\rho, 0]} (f_q^{-1}(y + \xi + \rho) - f^{-1}(x + \xi)), \\ X_R &= \{x \in [0, 1] : \exists y F_R(x, y) \leq 0\}, \\ X_L &= \{x \in [0, 1] : \exists y F_L(x, y) \leq 0\}, \\ Y_L &= \{y \in [0, 1] : \exists x F_L(x, y) \leq 0\}. \end{aligned}$$

Теорема 1. *Задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче одномерного интервального поиска тогда и только тогда, когда существуют функции $\psi, \psi_L, \psi_R : [0, 1] \rightarrow \mathbb{R}$ такие, что для любой пары (x, y) , для которой выполнено условие $F_L(x, y) \leq 0$, верно, что*

$$F_L(x, y) = \psi(y) + \psi_L(x), \quad (1)$$

а для любой пары (x, y) , такой, что $F_R(x, y) \leq 0$, верно, что

$$F_R(x, y) = \psi(y) + \psi_R(x). \quad (2)$$

При этом если выполнены условия (1) и (2) и

$$\begin{aligned}\psi^*(y) &= \begin{cases} \psi(y), & \text{если } y \in Y_L, \\ F_L(0, 1), & \text{если } y \notin Y_L, \end{cases} \\ \psi_R^*(x) &= \begin{cases} \psi_R(x), & \text{если } x \in X_R, \\ F_R(0, 1), & \text{если } x \notin X_R, \end{cases} \\ \psi_L^*(x) &= \begin{cases} \psi_L(x), & \text{если } x \in X_L, \\ F_L(0, 1), & \text{если } x \notin X_L, \end{cases}\end{aligned}$$

то функции сведения $(\varphi, \varphi_1, \varphi_2)$ имеют вид

$$\begin{aligned}\varphi(t_i, y_i) &= t_i - \psi^*(y_i), \\ \varphi_1(t_q, x) &= t_q + \psi_L^*(x), \\ \varphi_2(t_q, x) &= t_q + \psi_R^*(x).\end{aligned}$$

В разделе 3 построен алгоритм A_1 , решающий задачу о предотвращении столкновения сведением к задаче одномерного интервального поиска, и для этого алгоритма получены следующие оценки.

Теорема 2. Для алгоритма A_1 , решающего задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$ сведением к задаче одномерного интервального поиска верно, что для сложности поиска ответа на любой запрос q , вставки любого объекта-данного o , удаления любого объекта-данного \tilde{o} по алгоритму A_1 , и для объема алгоритма A_1 выполнены соответственно соотношения

$$\begin{aligned}T_{A_1}(f, f_q, \rho, V(t), q) &\leq c \log_2 |V|, \\ S_{A_1}(f, f_q, \rho, V(t), o) &\leq c \log_2 |V|, \\ R_{A_1}(f, f_q, \rho, V(t), \tilde{o}) &\leq c \log_2 |V|, \\ Q_{A_1}(f, f_q, \rho, V(t)) &\leq c|V|,\end{aligned}$$

где c — константа не зависящая от $|V|$.

Во второй главе приводится критерий сводимости задачи о предотвращении столкновений к одномерной задаче о прокалывании.

Задачей о прокалывании назовем пару (\mathbb{R}, Z) , где библиотека Z есть конечное множество всех интервалов с концами из \mathbb{R} , а \mathbb{R} есть множество всех действительных чисел. Содержательно эта задача состоит в том, чтобы для произвольного запроса $p \in \mathbb{R}$ перечислить все те и только те отрезки из Z , которые содержат p .

Ответ на запрос $p \in \mathbb{R}$ при библиотеке Z в задаче о прокалывании есть множество $J(p, Z) = \{z \in Z : p \in z\}$.

Будем говорить, что задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче о прокалывании, если существуют такие отображения $\varphi, \varphi_1, \varphi_2 : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}$, что для любой библиотеки $V(t)$, любого запроса $q = (t_q, x)$ и любого объекта $o \in V$ верно

$$o \in J(f, f_q, \rho, V, q) \Leftrightarrow [\varphi_1(o), \varphi_2(o)] \in J(\varphi(q), Z),$$

где $Z = \{[\varphi_1(o), \varphi_2(o)] : o \in V\}$.

Тройку $(\varphi, \varphi_1, \varphi_2)$ будем называть *функциями сведения*.

Аналогично критерию сводимости к задаче одномерного интервального поиска обозначим

$$Y_R = \{y \in [0, 1] : \exists x F_R(x, y) \leq 0\},$$

$$Y_L = \{y \in [0, 1] : \exists x F_L(x, y) \leq 0\},$$

$$X_L = \{x \in [0, 1] : \exists y F_L(x, y) \leq 0\}.$$

Теорема 3. *Задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче о прокалывании тогда и только тогда, когда существуют функции $\psi, \psi_L, \psi_R : [0, 1] \rightarrow \mathbb{R}$ такие, что для любой пары (x, y) , для которой выполнено условие $F_L(x, y) \leq 0$, верно, что*

$$F_L(x, y) = \psi(x) + \psi_L(y), \quad (3)$$

а для любой пары (x, y) , такой, что $F_R(x, y) \leq 0$, верно, что

$$F_R(x, y) = \psi(x) + \psi_R(y). \quad (4)$$

При этом если выполнены условия (3) и (4) и

$$\psi^*(x) = \begin{cases} \psi(x), & \text{если } x \in X_L, \\ F_L(0, 1), & \text{если } x \notin X_L, \end{cases}$$

$$\psi_R^*(y) = \begin{cases} \psi_R(y), & \text{если } y \in Y_R, \\ F_R(0, 1), & \text{если } y \notin Y_R, \end{cases}$$

$$\psi_L^*(y) = \begin{cases} \psi_L(y), & \text{если } y \in Y_L, \\ F_L(0, 1), & \text{если } y \notin Y_L, \end{cases}$$

то функции сведения $(\varphi, \varphi_1, \varphi_2)$ имеют вид

$$\varphi(t_q, x) = t_q + \psi^*(x),$$

$$\varphi_1(t_i, y_i) = t_i - \psi_R^*(y_i),$$

$$\varphi_2(t_i, y_i) = t_i - \psi_L^*(y_i).$$

В третьей главе рассматривается критерий сводимости задачи о предотвращении столкновений к задаче одномерного интервального поиска и задаче о прокалывании для случая законов движения объектов и запросов, являющихся аналитическими функциями. Формулировка критериев в этом случае значительно проще, чем в общем случае.

Теорема 4. Пусть законы движения f, f_q – аналитические функции. Тогда задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче одномерного интервального поиска или задаче о прокалывании тогда и только тогда, когда выполнено хотя бы одно из четырех условий

Условие 1.

1. Если $(x, y) \in [\rho, 1 + \rho] \times [0, 1] : f_q^{-1}(y) - f^{-1}(x) \leq 0$, то $(f_q^{-1}(y))' \leq (f^{-1}(x))'$;
2. Если $F_L(x, 0) \leq 0$, то $\rho \in \arg \min_{\xi \in [0, \rho]} F_L(x, 0, \xi)$;
3. Если $F_R(x, 1) \leq 0$, то $-\rho \in \arg \max_{\xi \in [-\rho, 0]} F_L(x, 1, \xi)$;
4. Если $(x, y) \in [0, 2\rho] \times [0, 1] : F_R(x, y) \leq 0$, то $-\rho \in \arg \max_{\xi \in [-\rho, 0]} F_R(x, y, \xi)$;
5. Если $(x, y) \in [0, \rho] \times [0, 1] : F_L(x, y) \leq 0$, то $\rho \in \arg \min_{\xi \in [0, \rho]} F_L(x, y, \xi)$;

Условие 2.

1. Если $(x, y) \in [0, 1] \times [-\rho, 1 - \rho] : f_q^{-1}(y) - f^{-1}(x) \leq 0$, то $(f_q^{-1}(y))' \geq (f^{-1}(x))'$;
2. Если $F_L(0, y) \leq 0$, то $0 \in \arg \min_{\xi \in [0, \rho]} F_L(0, y, \xi)$;
3. Если $F_R(1, y) \leq 0$, то $0 \in \arg \max_{\xi \in [-\rho, 0]} F_R(1, y, \xi)$;
4. Если $(x, y) \in [0, 1] \times [1 - \rho, 1] : F_R(x, y) \leq 0$, то $0 \in \arg \max_{\xi \in [-\rho, 0]} F_R(x, y, \xi)$;
5. Если $(x, y) \in [0, 1] \times [1 - \rho, 1] : F_L(x, y) \leq 0$, то $0 \in \arg \min_{\xi \in [0, \rho]} F_L(x, y, \xi)$;

Условие 3. f – линейный многочлен;

Условие 4. f_q – линейный многочлен.

При этом, в случаях 1, 3 задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ сводится к задаче о прокалывании, а в случаях 2, 4 к задаче одномерного интервального поиска.

В четвертой главе рассматриваются алгоритмы сокращения перебора. Идея этих алгоритмов состоит в следующем. Выбираются простые геометрические задачи поиска или их комбинация, ответ которых покрывает ответ задачи о предотвращении столкновений. Далее решаются вспомогательные геометрические задачи и в их ответе перебором находится ответ задачи о предотвращении столкновений.

Задачей о пересечении назовем пару (I, Z) , где библиотека Z есть конечное множество отрезков с концами из множества действительных чисел, а множество запросов I — есть множество всех отрезков с концами из множества действительных чисел. Содержательно эта задача состоит в том, чтобы для произвольного запроса $p \in I$ перечислить все те и только те отрезки из Z , которые пересекаются с отрезком p .

Ответ на запрос $p \in I$ при библиотеке Z в задаче о пересечении есть множество $J(p, Z) = \{z \in Z : z \cap p \neq \emptyset\}$.

Будем говорить, что задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ *накрывается задачей о пересечении*, если существуют такие отображения $\varphi_1, \varphi_2, \varphi_3, \varphi_4 : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}$, что для любой библиотеки $V(t)$, любого запроса $q = (t_q, x)$ и любого объекта $o \in V$ верно $o \in J(f, f_q, \rho, V, q) \Rightarrow [\varphi_1(o), \varphi_2(o)] \in J([\varphi_3(q), \varphi_4(q)], Z)$, где $Z = \{[\varphi_1(o), \varphi_2(o)] : o \in V\}$.

Четверку $(\varphi_1, \varphi_2, \varphi_3, \varphi_4)$ будем называть *функциями сведения*.

Предположим, что поток объектов $\mathbb{O} = \{o_1 = (t_1, y_1), o_2 = (t_2, y_2), \dots, o_i = (t_i, y_i), \dots\}$ представляет собой случайный процесс такой, что моменты появления объектов t_i образуют пуассоновский поток с параметром λ , а координаты появления y_i имеют равномерное и независимое распределение на отрезке $[0, 1]$ для каждого i . Такие потоки объектов будем называть *пуассоновскими потоками объектов*. Пусть $V(t)$ случайная библиотека в момент времени t , порожденная пуассоновским потоком объектов \mathbb{O} , т.е. $V(t) = \{o_i = (t_i, y_i) \in \mathbb{O} : t_i \leq t \leq t_i + \tau_{max}\}$

Средней сложностью поиска по алгоритму A на запросе q для расстояния опасной близости ρ назовем математическое ожидание по библиотеке от сложности поиска: $T_A(f, f_q, \rho, \lambda, q, t) = \mathbf{M}_V T_A(f, f_q, \rho, V(t), q)$, где математическое ожидание берется по всем случайным библиотекам $V(t)$, порожденными пуассоновскими потоками объектов с параметром λ , а $T_A(f, f_q, \rho, V(t), q)$ — сложность поиска по алгоритму A на запросе q .

Поскольку пуассоновский поток объектов является стационарным, то для достаточно больших t , а именно $t > \lambda \cdot \tau_{max}$, число объектов в библиотеке не зависит от t и приблизительно равно $\lambda \cdot \tau_{max}$. Отсюда в частности следует, что величина $T_A(f, f_q, \rho, \lambda, q, t)$ для достаточно больших

t не зависит от t , и поэтому будем писать $T_A(f, f_q, \rho, \lambda, q)$.

Будем писать $f(n) = O(g(n))$, если существуют такие константы C_1 и C_2 , что для любых $n \in \mathbb{N}$ выполнено $C_1 \cdot g(n) \leq f(n) \leq C_2 \cdot g(n)$. Введем следующие обозначения для задачи о предотвращении столкновений:

$$\begin{aligned} v_{min} &= \min_{y \in [-\rho, 1+\rho]} f'(y), & v_{max} &= \max_{y \in [-\rho, 1+\rho]} f'(y), \\ v_{min}^q &= \min_{y \in [-\rho, 1+\rho]} f'_q(y), & v_{max}^q &= \max_{y \in [-\rho, 1+\rho]} f'_q(y). \end{aligned}$$

Будем предполагать, что задача о пересечении может быть решена со средней сложностью $IS(\lambda\tau_{max})$, затрачивая объем памяти порядка $O(\lambda\tau_{max})$, где $\lambda\tau_{max}$ есть среднее число отрезков в библиотеке Z , совпадающее со средним числом объектов в области наблюдения. В этих предположениях построен алгоритм A_2 , который решает задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$.

Теорема 5. *Средняя сложность алгоритма A_2 , решающего задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$ сведением к задаче о пересечении, на любом запросе q удовлетворяет неравенствам:*

$$2\rho\lambda\left(\frac{1}{v_{max}} + \frac{1}{v_{max}^q}\right) \leq T_{A_2}(f, f_q, \rho, \lambda, q) - IS(\lambda\tau_{max}) \leq 2\rho\lambda\left(\frac{1}{v_{min}} + \frac{1}{v_{min}^q}\right),$$

а объем равен $Q_{A_2}(f, f_q, \rho, V(t)) = O(\lambda\tau_{max})$.

Будем говорить, что задача о предотвращении столкновений $(f, f_q, \rho, V(t))$ *накрывается двумя задачами одномерного интервального поиска и одной задачей о прокалывании*, если существуют отображения $\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6, \varphi_7, \varphi_8, \varphi_9 : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}$, что для любой библиотеки $V(t)$, любого запроса $q = (t_q, x)$ и любого объекта $o \in V$ верно, что если $o \in J(f, f_q, \rho, V, q)$, то либо $\varphi_1(o) \in J([\varphi_2(q), \varphi_3(q)], Z_1)$, где $Z_1 = \{\varphi_1(o) : o \in V\}$, либо $\varphi_4(o) \in J([\varphi_5(q), \varphi_6(q)], Z_2)$, где $Z_2 = \{\varphi_4(o) : o \in V\}$, либо $[\varphi_7(o), \varphi_8(o)] \in J([\varphi_9(q), Z_3)$, где $Z_3 = \{[\varphi_7(o), \varphi_8(o)] : o \in V\}$.

Будем предполагать, что задача о прокалывании (I_2, Z_2) может быть решена со средней сложностью $DO(\lambda\tau_{max})$, где $\lambda\tau_{max}$ есть среднее число отрезков в библиотеке Z_2 , совпадающее со средним числом объектов в области наблюдения. В этих предположениях построен алгоритм A_3 , который решает задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$.

Теорема 6. *Средняя сложность алгоритма A_3 , решающего задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$ сведением к двум задачам*

одномерного интервального поиска и одной задаче о прокалывании, на любом запросе q удовлетворяет неравенствам:

$$\begin{aligned} \frac{2\rho}{v_{max}}\lambda + O(\log_2(\lambda\tau_{max})) + DO(\lambda\tau_{max}) &\leq T_{A_3}(f, f_q, \rho, \lambda, q) \leq \\ &\leq \frac{2\rho}{v_{min}}\lambda + O(\log_2(\lambda\tau_{max})) + DO(\lambda\tau_{max}), \end{aligned}$$

а объем равен $Q_{A_3}(f, f_q, \rho, V(t)) = O(\lambda\tau_{max})$.

В четвертой главе также построен алгоритм A_4 , который решает задачу о предотвращении столкновений $(f, f_q, \rho, V(t))$ сведением к двум задачам о прокалывании и одной задаче одномерного интервального поиска.

В пятой главе рассматривается наиболее общий случай формулировки задачи о предотвращении столкновений.

Пусть $v_{min}, v_{max}, v_{min} < v_{max}$, — некоторые положительные числа, $\mathcal{F}_{v_{min}, v_{max}}$ — множество дифференцируемых строго возрастающих функций $f: \mathbb{R}^+ \rightarrow [-\rho, 1+\rho]$, таких что $f(0) = -\rho$, и $v_{min} \leq f'(\tau) \leq v_{max}$ для любого $\tau \in [0, f^{-1}(1+\rho)]$. Пусть на $\mathcal{F}_{v_{min}, v_{max}}$ задано некоторое вероятностное пространство.

Будем считать, что у каждого объекта o_i есть свой собственный закон движения $f_i \in \mathcal{F}_{v_{min}, v_{max}}$, и объект в этом случае представляет из себя тройку $o_i = (f_i, t_i, y_i)$, при этом как и ранее будем считать, что поток объектов $\mathbb{O} = \{o_1 = (f_1, t_1, y_1), o_2 = (f_2, t_2, y_2), \dots, o_i = (f_i, t_i, y_i), \dots\}$ представляет собой случайный процесс такой, что моменты появления объектов t_i образуют пуассоновский поток с параметром λ , координаты появления y_i имеют равномерное и независимое распределение на отрезке $[0, 1]$ для каждого i , а законы движения f_i выбираются в соответствии с распределением, заданным на $\mathcal{F}_{v_{min}, v_{max}}$, т.е. поток объектов является пуассоновским потоком. Пусть $V(t)$ случайная библиотека в момент времени t , порожденная пуассоновским потоком объектов \mathbb{O} , т.е. $V(t) = \{o_i = (f_i, t_i, y_i) \in \mathbb{O} : t_i \leq t \leq t_i + f_i^{-1}(1+\rho)\}$.

Поскольку пуассоновский поток объектов является стационарным, то для достаточно больших t , а именно $t > \lambda \cdot (1 + 2\rho)/v_{min}$, среднее число объектов в библиотеке не зависит от t . Отсюда в частности следует, что все сложностные характеристики алгоритмов для достаточно больших t не будут зависеть от t , и поэтому параметр t мы далее будем опускать, считая, что t достаточно большое.

Закон движения запроса f_q также не известен заранее и принадлежит множеству $\mathcal{F}_{v_{min}, v_{max}}$. Запрос в этом случае представляет из себя тройку $q = (f_q, t_q, x)$. Как и ранее, задача о предотвращении столкновений состоит в

том чтобы для произвольного, но зафиксированного пуассоновского потока объектов \mathbb{O} и произвольного запроса $q = (f_q, t_q, x)$ найти в библиотеке $V(t_q)$ все объекты, которые в процессе своего движения окажутся на расстоянии по Манхэттену не более чем ρ от запроса q . В такой формулировке задачу о предотвращении столкновений обозначим как пятерку $(v_{min}, v_{max}, \rho, \lambda, \mathbb{O})$.

Пусть имеется некоторый алгоритм A решения задачи о предотвращении столкновений и пусть $L_A(V, \dots)$ некоторая сложностная характеристика алгоритма A для библиотеки V . Тогда обозначим $L_A(v_{min}, v_{max}, \rho, \lambda, \dots) = \mathbf{M}_V L_A(V, \dots)$, где математическое ожидание берется по всем случайным библиотекам V , порожденными пуассоновскими потоками объектов с параметром λ и законами движения из $\mathcal{F}_{v_{min}, v_{max}}$.

В пятой главе приводится алгоритм A_5 решения задачи о предотвращении столкновений в общем случае.

Теорема 8. *Каково бы ни было вероятностное пространство над множеством законов движения $\mathcal{F}_{v_{min}, v_{max}}$, для алгоритма A_5 , решающего задачу о предотвращении столкновений $(v_{min}, v_{max}, \rho, \lambda, \mathbb{O})$, верно, что для произвольного запроса q средняя сложность поиска удовлетворяет неравенствам*

$$c_1 \sqrt{\lambda(1+2\rho)/v_{max}} \leq T_{A_5}(v_{min}, v_{max}, \rho, \lambda, q) \leq c_2 \sqrt{\lambda(1+2\rho)/v_{min}};$$

для любого объекта o средние сложности вставки и удаления удовлетворяют неравенствам

$$c_3 \sqrt{\lambda(1+2\rho)/v_{max}} \leq S_{A_5}(v_{min}, v_{max}, \rho, \lambda, o) \leq c_4 \sqrt{\lambda(1+2\rho)/v_{min}},$$

$$c_5 \sqrt{\lambda(1+2\rho)/v_{max}} \leq R_{A_5}(v_{min}, v_{max}, \rho, \lambda, o) \leq c_6 \sqrt{\lambda(1+2\rho)/v_{min}};$$

средний объем алгоритма A_5 удовлетворяет неравенствам

$$c_7 (\lambda(1+2\rho)/v_{max})^{3/2} \leq Q_{A_5}(v_{min}, v_{max}, \rho, \lambda) \leq c_8 (\lambda(1+2\rho)/v_{min})^{3/2},$$

где $c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$ — некоторые константы.

С учетом того, что среднее число объектов в библиотеке принадлежит отрезку $[\frac{\lambda}{v_{max}}(1+2\rho), \frac{\lambda}{v_{min}}(1+2\rho)]$, мы получаем, что основные средние сложностные характеристики алгоритма A_5 равны по порядку корню квадратному от мощности библиотеки, а объем памяти, необходимый для хранения структур данных, в среднем равен по порядку мощности библиотеки в степени $3/2$.

Благодарности

Автор выражает глубокую благодарность своему научному руководителю профессору Гасанову Э.Э. за постановку задачи и постоянное внимание к работе, академику Кудрявцеву В.Б. за многочисленные полезные советы на всех этапах подготовки диссертации и всему коллективу кафедры математической теории интеллектуальных систем Механико-математического факультета Московского государственного университета имени М.В.Ломоносова за теплую творческую атмосферу.

Публикации автора по теме диссертации

Из списка ВАК:

1. Скиба Е.А. Логарифмическое решение задачи об опасной близости // Интеллектуальные системы. 2007. 11: 1–4. С. 693–719.

2. Снегова Е.А. Случай задачи об опасной близости, сводящийся одномерному интервальному поиску // Интеллектуальные системы. 2009. 13: 1–4. С. 97–118.

3. Снегова Е.А. Критерий сводимости задачи об опасной близости к одномерному интервальному поиску // Дискретная математика. 2011. 23: 3. С. 138–158.

4. Снегова Е.А. Критерий сводимости задачи об опасной близости к одномерной задаче о прокалывании // Интеллектуальные системы. 2011. 15: 1–4. С. 281–306.

5. Снегова Е.А. Критерий сводимости задачи об опасной близости к одномерным задачам для полиномиальных законов движения // Интеллектуальные системы. 2011. 15: 1–4. С. 639–660.

6. Snegova, E. A. A criterion for reducibility of the problem on dangerous closeness to one-dimensional interval search // Discrete Mathematics and Applications. 2011. 21: 5–6. P. 701–725.

Публикации не из списка ВАК:

7. Скиба Е.А. Решение задачи об опасной близости при слабых ограничениях на законы движения // Международная конференция "Современные проблемы математики, механики и их приложений посвященная 70-летию ректора МГУ академика В.А.Садовниченко (30 марта - 2 апреля 2009 г., Москва). Материалы конференции. С. 374–375.

8. Скиба Е.А. Приближенное решение задачи об опасной близости // Сборник тезисов XVI Международной конференции студентов, аспирантов

и молодых ученых "Ломоносов-2009 секция "Математика и механика" (Москва, 13-18 апреля 2009). С. 61–62.

9. Снегова Е.А. Критерий сводимости задачи об опасной близости к одномерному интервальному поиску // Материалы X Международного семинара "Дискретная математика и ее приложения" (Москва, 1–6 февраля 2010 г.). Издательство механико-математического факультета МГУ, Москва, 2010. С. 432–434.

10. Снегова Е.А. О сводимости задачи об опасной близости к задаче одномерного интервального поиска // Тезисы докладов Секции "Математика и механика" Международной конференции студентов, аспирантов и молодых учёных "Ломоносов-2010". — М.: Механико-математический факультет МГУ имени М.В.Ломоносова, 2010.

11. Elena Snegova. Criteria for Reducibility of Moving Objects Closeness Problem // Advances in Databases and Information Systems. 2010. LNCS 6295. Pp. 583–586.

12. Снегова Е.А. Критерий сводимости задачи о предотвращении столкновений к задаче о прокалывании // Материалы X Международной конференции "Интеллектуальные системы и компьютерные науки" (5-10 декабря 2011 года). М.: 2011. С. 108–111.