Теория хранения и поиска информации

Э. Э. ГАСАНОВ
Московский государственный университет им. М. В. Ломоносова
e-mail: el_gasanov@mail.ru

УДК 519.95

Ключевые слова: хранение, поиск, информация, информационный поиск, база данных.

Аннотация

Построена новая общая модель хранения и поиска информации, называемая информационно-графовой, частными случаями которой являются известные модели представления данных. Изучены основные свойства этой модели и решена проблема оптимального синтеза информационных графов для широкого класса задач поиска, включающего наиболее часто используемые на практике задачи поиска в базах данных.

Abstract

E. E. Gasanov, Information storage and search complexity theory, Fundamentalnaya i prikladnaya matematika, vol. 15 (2009), no. 3, pp. 49—73.

We propose a new information-graph model for information storage and search. This model generalizes a number of known data representation models. We study the main properties of the proposed model. We solve the problem of optimal informational graph synthesis for a wide class of search problems including the most acute database search problems.

1. Введение

В последние десятилетия активно развивается новое научное направление, связанное с оптимальным хранением и поиском информации, именуемое теорией информационного поиска. Важной составляющей в нём является теория баз данных. Возникшее под влиянием практических задач, оно и сейчас в основном обслуживает приложения, а собственно теоретическая его часть, как представляется, обретает контуры. Как всякая научная дисциплина, это направление должно характеризоваться следующими чертами: предметом исследования, проблематикой, методами и результатами. В развитой теории каждая из этих черт должна иметь достаточно общий характер. В то же время важно отметить, что молодые дисциплины возникают, как правило, через рассмотрение отдельных

Фундаментальная и прикладная математика, 2009, том 15, № 3, с. 49—73.

конкретных важных примеров, которые затем, с развитием дисциплин, обобщаются как в постановочной, так и в проблемно-методологической частях. Подобных примеров достаточно много в кибернетике, информатике и других разделах науки. К числу характерных может быть отнесена теория управляющих систем. В своей практической деятельности человек столкнулся с конкретными видами таких систем, которые далее играли роль модельных управляющих систем. В инженерном деле это вентильные и контактные схемы, схемы из функциональных элементов и некоторые другие, в математике — формулы, алгоритмы и т. д., в биологии — нейроны, нейронные сети, автоматы и т. п. Для этих видов управляющих систем рассматривались следующие две главные задачи анализа и синтеза соответствующих управляющих систем [36, 52]. Первая состояла в изучении «поведения» таких систем, а вторая — в создании соответствующей системы с заданным «поведением». На первом этапе эти постановки связывались непосредственно с модельными системами, и для каждой из них разрабатывался конкретный метод решения. Со временем наступил этап, когда большинство из этих систем могли уже рассматриваться с единых позиций, и исследование указанных общих задач достигалось уже с помощью общих методов решения [1]. Хотя по-прежнему модельные управляющие системы, имея свою специфику, продолжают оставаться в центре внимания теории управляющих систем.

Аналогичный путь развития проходит теория информационного поиска. В ней также первоначально возникли конкретные примеры способов хранения и представления данных и соответствующих этим способам алгоритмов поиска информации: лексикографические, древовидные, реляционные и др. Задачи поиска для них имеют конкретные виды и модификации: например, задача поиска идентичных объектов, задача о близости, включающий поиск и др. Они играют роль модельных задач для выбранных способов хранения информации и изучались на протяжении многих лет. Каждый раз для их решения привлекались специальные исследовательские средства, которые носили ограниченный по своим возможностям характер.

Таким образом, можно считать, что современное состояние теории информационного поиска напоминает то состояние теории управляющих систем, которое соответствовало первому этапу развития последней, когда накапливались данные лишь о конкретных видах модельных управляющих систем. В то же время, как выяснилось, опыт развития теории управляющих систем в своей методологической части даёт возможность сделать попытку с более общих позиций провести исследование как модельных баз данных, так и модельных задач для них, с соответствующей разработкой достаточно общей теории.

Мы предлагаем новую модель данных (частными случаями которой могут считаться уже известные) с наследственно определёнными средствами поиска информации и соответствующими понятиями сложности такого поиска, а также разрабатываем основы теории решения базовых задач поиска применительно к этой модели. Если продолжить аналогию с теорией синтеза управляющих систем, то можно отметить, что различным видам управляющих систем соответствуют различные виды хранения и представления данных (модели данных),

классам функций, исследуемым в теории синтеза, соответствуют типы задач поиска, исследуемые в теории информационного поиска. И в теории синтеза, и в теории поиска вводятся понятия сложности и ставятся задача оптимального синтеза и задача исследования функций сложности шенноновского типа. Таким образом, мы стремимся к тому, чтобы приблизить состояние теории информационного поиска по степени продвинутости к современному состоянию теории управляющих систем.

Уточним сказанное. Во-первых, мы предлагаем новую формализацию понятия задачи поиска. Тип задач поиска охватывает класс однотипных вопросов к базе данных. Тип задач поиска заключает в своём определении три объекта: множество запросов, множество записей и бинарное отношение, заданное на декартовом произведении этих множеств, называемое отношением поиска. Здесь запись — это поисковый образ элемента данных, т. е. поле или множество полей элемента данных, которые представляют интерес в данном типе вопросов. Запрос — это минимальный элемент, содержащий суть вопроса. Запрос вместе с отношением очерчивает тот круг объектов, которые отвечают на данный вопрос. Задача поиска заданного типа получается выделением из множества записей конечного подмножества, называемого библиотекой. А именно, задача поиска состоит в том, чтобы по произвольному запросу перечислить все записи из библиотеки, находящиеся в заданном отношении с запросом (удовлетворяющие запросу). При фиксации отношения поиска каждая запись задаёт предикат, определённый на множестве запросов, который равен 1, если данная запись удовлетворяет запросу — аргументу функции. Поэтому если вернуться к аналогии с теорией синтеза управляющих систем, то тип задач поиска есть способ описания некоторого конкретного класса предикатов, задаваемых на множестве запросов, а задача поиска — это конкретное подмножество предикатов из этого класса.

Во-вторых, мы предлагаем новую управляющую систему, называемую информационным графом, которая в общей иерархии теории управляющих систем находится в не очень высоких слоях залегания и является в некотором смысле обобщением контактных схем. Фактически, нам нужны лишь графы, дискретные функции и вычисление волновых процессов на графах, и этого хватает, чтобы с достаточно общих позиций посмотреть на ту разрозненную картину, которая наблюдается в теории информационного поиска.

В предлагаемой информационно-графовой модели данных структура данных задаётся ориентированным графом (называемым информационным), рёбра и вершины которого нагружены элементами данных и функциями, определёнными на множестве запросов. В графе выделена одна вершина, называемая корнем и ассоциируемая со входом, а вершины графа, нагруженные элементами данных, ассоциируются с выходами. Этот же граф описывает алгоритм поиска, на вход которого поступает запрос, а на выходе получается некоторое подмножество данных. При этом процесс поиска начинается с корня и распространяется в зависимости от значений нагрузочных функций на запросе, возможно сразу по нескольким направлениям. Если этот волновой процесс на графе достигает эле-

ментов данных, то эти элементы включаются в ответ алгоритма на исходный запрос. Информационный граф будет решать некоторую задачу поиска, если для произвольного запроса ответ на этот запрос содержит все те и только те записи из библиотеки, которые удовлетворяют запросу. Таким образом, информационный граф, с одной стороны, даёт новую концепцию хранения данных, а с другой стороны, предоставляет новый подход к поиску информации, заключающийся в использовании волнового процесса на графах, управляемого нагрузочными функциями. Нагрузочные функции, которые называются базовыми, разделены на два класса — предикаты и переключатели, и являются одними из основных управляющих параметров модели. Нагрузочные функции, по сути, определяют функции проводимости между вершинами графа, и проблема нахождение решения задачи поиска сводится к проблеме синтеза информационного графа, реализующего систему функций, задаваемую задачей поиска.

Как логическая сеть со свободными элементами [29] обобщает известные в теории управляющих систем виды управляющих систем, так и информационно-графовая модель обобщает наиболее известные модели данных. Понятно, что алгоритмы и конструкции, используемые в *древовидных базах данных*, описываются древовидными информационными графами. *Сетевые базы данных* естественным образом перекладываются на язык информационно-графовой модели, при этом ясно, что со структурной точки зрения они, по существу, будут представляться графами. В *дедуктивных базах* нужные данные и знания получаются путём логического вывода, поэтому алгоритм поиска, используемый в дедуктивных базах данных, при переходе на язык информационных графов приводит к константному дереву, который отражает суть дерева логического вывода. В *реляционных базах* данные представляются в виде таблиц, при этом алгоритм поиска, ассоциируемый с таким представлением данных, есть алгоритм перебора, который легко описывается древовидным информационным графом специального вида.

Информационные графы позволяют ввести новое понятие сложности поиска. Это понятие новое как с точки зрения теории управляющих систем, так и с точки зрения теории баз данных. В теории управляющих систем обычно под сложностью понимается или число рёбер, или число элементов-функций, а здесь сложность понимается как часть графа, захваченного волновым процессом. Она существенно зависит от значений нагрузочных функций и тем самым не является просто количественной характеристикой графа, такой как число рёбер или вершин. Новизна же в теории баз данных заключается в том, что такое введение сложности после осреднения по множеству запросов адекватно соответствует среднему времени поиска — традиционно трудной для изучения характеристики алгоритмов поиска информации. Кроме того, при соответствующем введении сложности информационные графы оказываются удобными для изучения как параллельных, так и фоновых алгоритмов поиска. Наконец, в информационных графах совсем просто контролируется такой важный управляющий параметр в задачах информационного поиска, как объём памяти, который в данном случае характеризуется количеством рёбер графа.

Рассматривается несколько модельных типов задач поиска, являющихся наиболее распространёнными задачами поиска в базах данных. Выбор модельных типов определяется как повсеместностью использования их в базах данных, так и частотой цитирования в литературе [26,28,29,35,41,42,46—48,55—62,64] Эти модельные типы можно разбить на три крупных базовых класса.

Первый класс включает в себя задачи поиска, в которых для почти всех запросов ответ содержит ограниченное малой константой число элементов. Этот класс получил название задач поиска с коротким ответом. Представителем этого класса является задача поиска идентичных объектов.

Второй класс — задачи поиска на частично упорядоченных множествах данных — состоит из задач, в которых в ответ на запрос надо перечислить все элементы базы данных, которые в заданном частичном порядке меньше, чем запрос. Представителями этого класса являются задача включающего поиска и задача о доминировании.

Наконец, третий класс содержит так называемые задачи интервального поиска, результат которых в некотором смысле можно рассматривать как пересечение решений двух задач из второго класса.

Для базовых задач поиска ставится и решается проблема оптимального синтеза, которая состоит в построении для заданной задачи информационного поиска информационного графа, который решает эту задачу и имеет наименьшую или близкую к ней сложность.

Полученный свод результатов, описывающих оптимальное решение для базовых классов, назовём каноническим эффектом. Мы хотим понять, насколько чувствительна основная модель по отношению к каноническому эффекту при вариации трёх основных управляющих параметров модели, таких как объём памяти, имеющийся в распоряжении (т. е. число рёбер информационного графа), множество функций, которые разрешается использовать при решении (т. е. множество базовых функций, используемых при нагрузке графа), и ε-расширение запроса. Показывается, что при любой вариации, кроме ε-расширения запроса при достаточно малых ε, мы уходим от канонического эффекта.

Для решения задач оптимального синтеза для базовых классов разработаны следующие три основных метода.

Первый метод мы называем методом оптимальной декомпозиции. Он состоит в таком разбиении задачи на подзадачи, которые допускают простое решение. При этом поиск подзадач, дающих решение исходной задачи, также осуществляется просто. Этот метод использовался при решении опорных или одномерных задач поиска.

Второй метод, называемый методом снижения размерности, применяемый к многомерным задачам, сводится к тому, чтобы с помощью некоторых опорных задач последовательно понижать размерность задачи и в конце концов свести её к опорной задаче, решение которой уже известно.

Третий метод назван методом характеристических носителей графа и использовался при получении нижних оценок. Он заключается в выделении в информационном графе, являющемся оптимальным решением, подграфов с заданными

свойствами (характеристических носителей) и в последующем подсчёте сложности характеристических носителей.

Данная работа содержит обзор результатов автора, полученных им в этом направлении.

Автор выражает глубокую благодарность академику В. Б. Кудрявцеву и профессору А. С. Подколзину за внимание и помощь в работе.

2. Информационно-графовая модель данных

В задачах поиска, возникающих в базах данных, имеется три основных объекта:

- *множество запросов* X с заданным на нём вероятностным пространством;
- множество потенциальных ответов Y (мы будем называть элементы этого множества *записями*);
- бинарное отношение ρ, заданное на $X \times Y$, называющееся отношением поиска и описывающее критерий семантического соответствия записи запросу, т. е. если $x \ \rho \ y$, то будем говорить, что запись y удовлетворяет запросу x.

В достаточно общем случае значительный интерес представляет описываемая ниже проблема, которую мы назовём задачей информационного поиска. Тройку $\langle X, Y, \rho \rangle$ будем называть *типом задач информационного поиска*, а тройку $\langle X, V, \rho \rangle$ (или четвёрку $\langle X, V, \rho; Y \rangle$), где V — конечное подмножество Y, называемое *библиотекой*, — *задачей информационного поиска*. Содержательно будем считать, что задача информационного поиска $I = \langle X, V, \rho; Y \rangle$ состоит в перечислении для произвольно взятого запроса $x \in X$ всех тех и только тех записей из V, которые находятся в отношении ρ с запросом x, т. е. удовлетворяют запросу x.

Эта проблема допускает вариацию как за счёт уточнения самой задачи, так и за счёт разных предположений относительно базовых компонент X, Y, ρ, V, составляющих задачу информационного поиска.

Опишем основной объект, который называется информационным графом. Вводить информационный граф мы будем, одновременно иллюстрируя его на примере одномерной задачи интервального поиска, которая состоит в поиске в конечном подмножестве отрезка $[0, 1]$ вещественной прямой всех тех точек, которые попадают в отрезок-запрос.

Сначала задаются четыре множества:

- множество запросов X;
- множество записей Y;
- множество F одноместных предикатов, заданных на множестве X;
- множество G одноместных переключателей, заданных на множестве X (переключатели — это функции, область значений которых является начальным отрезком натурального ряда).

В примере эти множества имеют вид

- $X_{\text{int1}} = \{(u, v)\colon 0 < u \leqslant v \leqslant 1\}$;
- $Y_{\text{int1}} = (0, 1]$;
- $F = F_1 \cup F_2$, где $F_1 = \{f^1_{\leqslant,a}\colon a \in (0,1]\}$, $F_2 = \{f^2_{\geqslant,a}\colon a \in (0,1]\}$,

$$f^1_{\leqslant,a}(u,v) = \begin{cases} 1, & \text{если } u \leqslant a, \\ 0, & \text{если } u > a, \end{cases} \qquad f^2_{\geqslant,a}(u,v) = \begin{cases} 1, & \text{если } v \geqslant a, \\ 0, & \text{если } v < a; \end{cases}$$

- $G = G_1 \cup G_2 \cup G_3$, где $G_1 = \{g_{*,m}\colon m \in \mathbb{N}\}$, $G_2 = \{g_{-,m}\colon m \in \mathbb{N}\}$, $G_3 = \{g_{\leqslant,a}\colon a \in (0,1]\}$, $g_{*,m}(u,v) =]u \cdot m[$,

$$g_{-,m}(u,v) = \begin{cases} 1, & \text{если } v - u < 1/m, \\ 2, & \text{если } v - u \geqslant 1/m, \end{cases} \qquad g_{\leqslant,a}(u,v) = \begin{cases} 1, & \text{если } u \leqslant a, \\ 2, & \text{если } u > a. \end{cases}$$

Информационный граф определяется следующим образом. Берётся конечная многополюсная ориентированная сеть. В ней выбирается некоторый полюс, который называется корнем. На рис. 1 он изображён полым кружком. Остальные полюса называются листьями (на рисунке они изображены жирными точками), им приписываются записи из Y (на рисунке это символы y с индексами), причём разным листьям могут быть приписаны одинаковые записи. Некоторые вершины сети (это могут быть и полюса) называются переключательными, им приписываются переключатели из G (на рисунке таких вершин 8). Рёбра, исходящие из

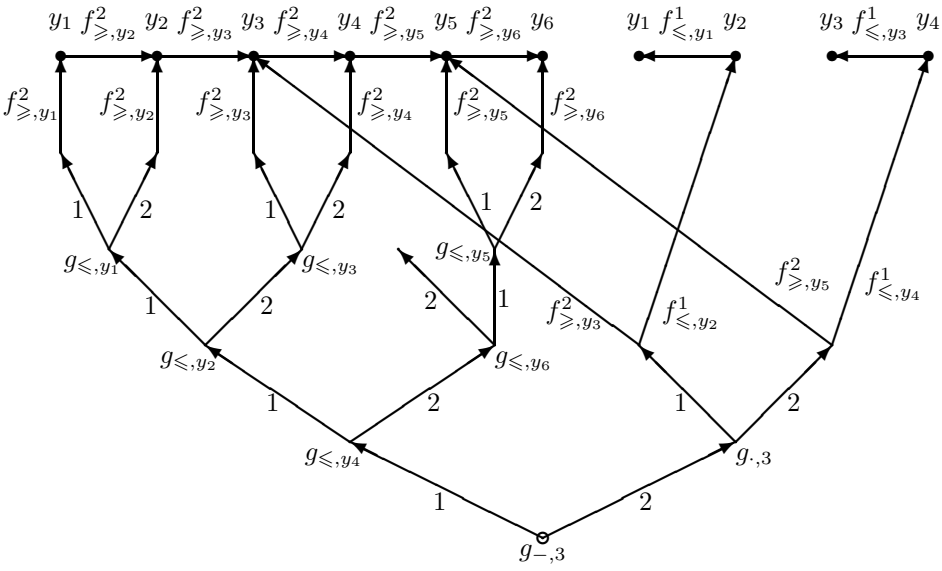

Рис. 1. Решение одномерной задачи интервального поиска

каждой из переключательных вершин, нумеруются начиная с 1 и называются переключательными рёбрами (на рисунке таких рёбер 16). Рёбра, не являющиеся переключательными, называются предикатными, им приписываются предикаты из множества F (на рисунке таких рёбер 17). Таким образом нагруженную многополюсную ориентированную сеть называем информационным графом над базовым множеством $\mathcal{F} = \langle F, G \rangle$.

Функционирование информационного графа определяется следующим образом. Скажем, что предикатное ребро проводит запрос $x \in X$, если предикат, приписанный этому ребру, принимает значение 1 на запросе x. Скажем, что переключательное ребро, которому приписан номер n, проводит запрос $x \in X$, если переключатель, приписанный началу этого ребра, принимает значение n на запросе x. Скажем, что ориентированная цепочка рёбер проводит запрос $x \in X$, если каждое ребро цепочки проводит запрос x. Скажем, что запрос $x \in X$ проходит в вершину β информационного графа, если существует ориентированная цепочка, ведущая из корня в вершину β, которая проводит запрос x. Скажем, что запись y, приписанная листу α, попадает в ответ информационного графа на запрос $x \in X$, если запрос x проходит в лист α. Ответом информационного графа U на запрос x назовём множество записей, попавших в ответ информационного графа на запрос x; обозначим его $\mathcal{J}_U(x)$. Функцию $\mathcal{J}_U(x)$ будем считать результатом функционирования информационного графа U.

Из определения функционирования информационного графа естественным образом вытекает, что каждому информационному графу U можно поставить в соответствие некую процедуру поиска. Предполагается, что эта процедура хранит в своей (внешней) памяти структуру информационного графа U. Входные данные процедуры — запрос. Выходные данные — множество записей.

Опишем эту процедуру. Пусть на вход процедуры поступил запрос x. Введём понятие активного множества вершин, внесём в него в начальный момент корень информационного графа U, пометим его. Далее по очереди просматриваем вершины из активного множества и для каждой из них проделываем следующее:

- если рассматриваемая вершина — лист, то запись, приписанную вершине, включаем в ответ;
- если рассматриваемая вершина переключательная, то вычисляем на запросе x переключатель, соответствующий данной вершине; если конец ребра, исходящего из рассматриваемой вершины, нагрузка которого равна значению переключателя, — непомеченная вершина, то помечаем эту вершину и включаем в множество активных вершин;
- если рассматриваемая вершина предикатная, то просматриваем по очереди исходящие из неё ребра и вычисляем значения предикатов, приписанных этим ребрам, на запросе x. Концы рёбер, которым соответствуют предикаты со значениями, равными 1, если они непомеченные, помечаем и включаем в множество активных вершин;
- исключаем рассматриваемую вершину из активного множества.

Процедура завершается по исчерпании активного множества.

Таким образом, информационный граф как управляющая система может рассматриваться в качестве модели алгоритма поиска, работающего над данными, организованными в структуру, определяемую структурой информационного графа.

Пусть нам дана задача информационного поиска $I = \langle X, V, \rho \rangle$. Скажем, что информационный граф U *разрешает* задачу информационного поиска $I = \langle X, V, \rho \rangle$, если для любого запроса $x \in X$ ответ на этот запрос содержит все те и только те записи из V, которые удовлетворяют запросу x, т. е.

$$\mathcal{J}_U(x) = \{y \in V : x \ \rho \ y\}.$$

Если ρ_{int1} — бинарное отношение на $X_{\text{int1}} \times Y_{\text{int1}}$, такое что

$$(u, v) \ \rho_{\text{int1}} \ y \iff u \leqslant y \leqslant v,$$

то информационный граф, изображённый на рис. 1, разрешает задачу информационного поиска $I = \langle X_{\text{int1}}, V, \rho_{\text{int1}} \rangle$, где $V = \{y_1, y_2, y_3, y_4, y_5, y_6\}$ — библиотека, изображённая на рис. 2, причём данный информационный граф соответствует асимптотически оптимальному решению, полученному по методу оптимальной декомпозиции, описание которого применительно к данной задаче мы приведём ниже.

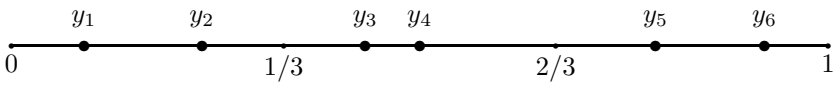

Рис. 2

Введём вспомогательные обозначения. Если f — одноместный предикат, определённый на X, то множество $N_f = \{x \in X : f(x) = 1\}$ назовём *характеристическим множеством* предиката f. Множество $O(y, \rho) = \{x \in X : x \ \rho \ y\}$ назовём *тенью* записи $y \in Y$.

Введём понятие сложности информационного графа.

Пусть β — некоторая вершина информационного графа. Предикат, определённый на множестве запросов, который принимает значение 1 на запросе x, если запрос проходит в вершину β, и 0 в противном случае, назовём функцией фильтра вершины β и обозначим $\varphi_\beta(x)$.

Определим понятие сложности информационного графа на запросе.

Будем считать, что время вычисления любого переключателя из G и любого предиката из F одинаково и равно 1.

Пусть нам дан некий информационный граф U и произвольно взятый запрос $x \in X$. Пусть A — определённая ранее процедура, соответствующая информационному графу U. *Сложностью информационного графа U на запросе x* назовём число $T(U, x)$, равное количеству переключателей и предикатов, вычисленных процедурой A при подаче на его вход запроса x, т. е.

$$T(U, x) = \sum_{\beta \in \mathcal{P}} \varphi_\beta(x) + \sum_{\beta \in \mathcal{R} \setminus \mathcal{P}} \psi_\beta \cdot \varphi_\beta(x),$$

где $\mathcal{R}$ — множество вершин графа U, $\mathcal{P}$ — множество переключательных вершин графа U, ψ_β — количество рёбер, исходящих из вершины β. Величина $T(U,x)$ характеризует время работы процедуры A при подаче на вход графа запроса x.

Введём понятие сложности информационного графа как среднее значение сложности информационного графа на запросе, взятое по множеству всех запросов. С этой целью введём *вероятностное пространство* над множеством запросов X, под которым будем понимать тройку $\langle X,\sigma,\mathbf{P}\rangle$, где σ — некоторая *алгебра подмножеств* множества X, $\mathbf{P}$ — *вероятностная мера* на σ, т. е. аддитивная мера, такая что $\mathbf{P}(X)=1$.

Скажем, что базовое множество $\mathcal{F}$ *измеримое*, если каждая функция из $\mathcal{F}$ измеримая (относительно алгебры σ). Далее всюду будем предполагать, что базовое множество измеримое. В этом случае для любого информационного графа U над $\mathcal{F}$ функция $T(U,x)$ как функция от x измерима.

Сложностью информационного графа U назовём математическое ожидание величины $T(U,x)$, т. е. число

$$T(U)=\mathbf{M}_{\mathrm{x}}\,T(U,x).$$

Объёмом $Q(U)$ информационного графа U назовём число рёбер в информационном графе U.

Пусть нам дана некая задача информационного поиска I. *Сложностью задачи* I *при базовом множестве* $\mathcal{F}$ *и заданном объёме* q назовём число

$$T(I,\mathcal{F},q)=\inf\{T(U)\colon U\in\mathcal{U}(I,\mathcal{F}),\ Q(U)\leqslant q\},$$

где $\mathcal{U}(I,\mathcal{F})$ — множество всех информационных графов над базовым множеством $\mathcal{F}$, разрешающих задачу информационного поиска I.

Число

$$T(I,\mathcal{F})=\inf\{T(U)\colon U\in\mathcal{U}(I,\mathcal{F})\}$$

назовём *сложностью задачи* I *при базовом множестве* $\mathcal{F}$.

Если $I=\langle X,V,\rho\rangle$, то величина

$$R(I)=\sum_{y\in V}\mathbf{P}\big(O(y,\rho)\big)$$

есть средняя длина ответа задачи информационного поиска I.

Теорема 1 (мощностная нижняя оценка [12]). *Если* $I=\langle X,V,\rho\rangle$ *— произвольная задача информационного поиска,* $\mathcal{F}$ *— измеримое базовое множество, такое что множество* $\mathcal{U}(I,\mathcal{F})\neq\varnothing$*, то* $T(I,\mathcal{F})\geqslant R(I)$.

Этот результат был получен с помощью метода характеристических носителей графа.

3. Решение проблемы оптимального синтеза для базовых задач

Если существует такой информационный граф $U \in \mathcal{U}(I, \mathcal{F})$, что $T(U) = T(I, \mathcal{F})$, то информационный граф U будем называть *оптимальным* для задачи информационного поиска I.

Для модельных классов ставится проблема синтеза оптимального информационного графа.

Среди *задач поиска, в которых вероятность появления в ответе более c записей* ($c = \mathrm{const}$) *равна нулю*, наиболее подробно исследована ситуация, когда $c = 1$. Для таких задач показано, что оптимальные информационные графы древовидны, а в случае когда тени всех записей библиотеки имеют равную вероятность (такие задачи информационного поиска названы обладающими G-свойством), справедлив следующий результат [13].

Теорема 2. *Если $I = \langle X, V, \rho\rangle$ — задача информационного поиска, обладающая G-свойством, $\mathcal{F} = \langle F, \varnothing\rangle$ — некоторое специальное базовое множество, то*

$$\mathbf{P}\big(O(y,\rho)\big) \cdot R(k) \leqslant T(I,\mathcal{F}) \leqslant \mathbf{P}\big(O(y,\rho)\big) \cdot R(k) + 1,$$

где $y \in V$, $k = |V|$ — мощность библиотеки V,

$$R(k) = 3k[\log_3 k] + 4\big(k - 3^{[\log_3 k]}\big) + \max\big(0, k - 2\cdot 3^{[\log_3 k]}\big).$$

Здесь и далее формулировки теорем носят несколько упрощённый характер и служат только для того, чтобы отразить общую картину. Строгие формулировки можно найти по ссылкам. Этот результат был получен методом характеристических носителей графа.

Задача *поиска идентичных объектов* состоит в поиске в множестве объекта, идентичного объекту-запросу, и формально принадлежит типу

$$S_{\mathrm{id}} = \langle (0,1], (0,1], =, \sigma, \mathbf{P}\rangle.$$

Задача о близости состоит в поиске в линейно-упорядоченном множестве объекта, ближайшего к объекту-запросу, и принадлежит типу

$$S_{\mathrm{ne}} = \langle (0,1], (0,1], \rho_{\mathrm{ne}}, \sigma, \mathbf{P}\rangle,$$

где ρ_{ne} задаётся на $(0,1] \times V$ и определяется соотношением

$$x\ \rho_{\mathrm{ne}}\ y \iff (y \in V)\ \&\ (x \leqslant y)\ \&\ \Big(\neg(\exists\, y')\big((y' \in V)\ \&\ (x \leqslant y')\ \&\ (y' < y)\big)\Big).$$

Пусть

$$F_3 = \left\{ f_{=,a}(x) = \begin{cases} 0, & \text{если } x \neq a, \\ 1, & \text{если } x = a: \end{cases} \quad a \in (0,1] \right\}, \tag{1}$$

$$G_4 = \left\{ g_{\leqslant,a}(x) = \begin{cases} 1, & \text{если } \ x \leqslant a, \\ 2 & \text{в противном случае:} \end{cases} \qquad a \in (0,1] \right\}, \tag{2}$$

$$G_5 = \{]x \cdot m[\colon m = 1,2,3\ldots\}, \quad \mathcal{F} = \langle F_3,\, G_4 \cup G_5 \rangle. \tag{3}$$

Теорема 3 [15]. *Пусть вероятностная мера* $\mathbf{P}$ *определяется ограниченной константой* c *функцией плотности распределения,* I *— задача информационного поиска типа* S_{id} *или типа* S_{ne}*,* $\mathcal{F}$ *— базовое множество, задаваемое соотношениями* (1)—(3)*. Тогда*

$$1 < T(I, \mathcal{F}, (2+c) \cdot k + 1) < 2.$$

Эта теорема получена методом оптимальной декомпозиции.

В [24] для задачи поиска идентичных объектов приводится алгоритм, который в типичной ситуации при затратах памяти k^2 обеспечивает время поиска, в худшем случае равное 2.

Задача информационного поиска с отношением поиска, являющимся отношением линейного предпорядка, — первая из задач, относящихся ко второму классу задач поиска на частично-упорядоченных множествах данных. Отношение линейного предпорядка — это отношение, удовлетворяющее условиям рефлексивности, транзитивности и связности.

Будем рассматривать следующий тип задач информационного поиска:

$$S_{\mathrm{lin}} = \left\langle X, X, \overset{1}{\succeq} \right\rangle,$$

где X — некоторое множество, $\overset{1}{\succeq}$ — некоторое отношение линейного предпорядка на $X \times X$. Пусть

$$\mathcal{K} = \left\{ \chi_{a,\overset{1}{\succeq}}(x) \colon a \in X \right\}.$$

Теорема 4 [11]. *Для любой задачи информационного поиска* $I = \left\langle X, V, \overset{1}{\succeq} \right\rangle$ *типа* S_{lin} *существует оптимальный информационный граф над базовым множеством* $\mathcal{F} = \langle \mathcal{K}, \varnothing \rangle$ *и*

$$T(I, \mathcal{F}) = 1 + R(I) - \min_{y \in V} \mathbf{P}\left(O\left(y, \overset{1}{\succeq}\right)\right).$$

Для задачи информационного поиска типа S_{lin} исследовалось также параллельное решение [20], которое предполагает, что информационный граф обрабатывается сразу несколькими вычислителями. При этом выделяется два подхода: когда информационный граф распределяется на части между вычислителями и каждый вычислитель обрабатывает только свою часть (сепаративный подход) и когда вычислители совместно обрабатывают информационный граф (кооперативный подход). Получено оптимальное параллельное решение в случае сепаративного подхода и показано существование таких задач информационного поиска типа S_{lin}, для которых кооперативный подход даёт лучшие результаты, чем сепаративный подход.

Задача *включающего поиска* принадлежит следующему типу:

$$S_{\mathrm{bool}} = \left\langle B^n, B^n, \overset{\mathrm{b}}{\succeq} \right\rangle,$$

где B^n — *единичный n-мерный куб*, $\overset{\mathrm{b}}{\succeq}$ — отношение поиска на $B^n \times B^n$, определяемое следующим соотношением:

$$(x_1, \ldots, x_n) \overset{\mathrm{b}}{\succeq} (y_1, \ldots, y_n) \Longleftrightarrow x_i \geqslant y_i,\ i = 1, 2, \ldots, n,$$

причём на B^n задана равномерная вероятностная мера, т. е. для всех $x \in B^n$ выполнено $\mathbf{P}(x) = 1/2^n$ и для всех $A \subseteq B^n$ выполнено $\mathbf{P}(A) = |A|/2^n$.

Теорема 5 [16]. *Пусть базовое множество имеет вид $\mathcal{F} = \langle F, \varnothing \rangle$, где $F \subseteq \mathcal{M}^n$ и $\mathcal{K}^n \subseteq F$, и $\mathcal{M}^n$ — множество монотонных булевых функций, а $\mathcal{K}^n$ — множество элементарных монотонных конъюнкций. Тогда для любой задачи информационного поиска $I = \left\langle B^n, V, \overset{\mathrm{b}}{\succeq} \right\rangle$ типа S_{bool}*

$$T(I, \mathcal{F}) \geqslant 2R(I)$$

и существуют такие задачи информационного поиска $I = \left\langle B^n, V, \overset{\mathrm{b}}{\succeq} \right\rangle$ типа S_{bool}, что

$$T(I, \mathcal{F}) = 2R(I)\big(1 + o(1)\big) \quad \textit{при} \quad n \to \infty.$$

Нижняя оценка этой теоремы была получена с помощью метода характеристических носителей графа. Приведём краткое описание этого метода применительно к задаче включающего поиска. На первом этапе показывается, что для каждой записи из библиотеки задачи в информационном графе, решающем данную задачу, существует так называемая главная цепь, т. е. цепочка рёбер, ведущая из корня информационного графа в лист, которому приписана данная запись, и по этой цепочке проходят все запросы, которым удовлетворяет данная запись. Далее путём перебора различных вариантов пересечения главных цепей показывается, что библиотеку можно разбить на непересекающиеся части таким образом, что каждой части можно поставить в соответствие своё подмножество рёбер графа (такие подмножества обычно имеют вид метёлки), суммарная сложность которых не меньше чем удвоенная сумма вероятностей теней записей из данной части.

Как видно, теорема 5 даёт асимптотику функции Шеннона. Кроме того, для включающего поиска была получена асимптотика логарифма сложности для почти всех задач и для средней сложности по задачам.

Задача о доминировании состоит в поиске в конечном подмножестве n-мерного пространства всех тех точек, которые по каждой из компонент не больше, чем запрос, являющийся в данном случае точкой n-мерного пространства. Пусть $X_{\mathrm{dom}} = (0, 1]^n$. Отношение поиска ρ_{dom} определено на $X_{\mathrm{dom}} \times X_{\mathrm{dom}}$ и задаётся следующим соотношением:

$$(x_1, x_2, \ldots, x_n)\ \rho_{\mathrm{dom}}\ (y_1, y_2, \ldots, y_n) \Longleftrightarrow y_i \leqslant x_i,\ i = 1, 2, \ldots, n.$$

Тогда тип

$$S_{\mathrm{dom}} = \langle X_{\mathrm{dom}}, X_{\mathrm{dom}}, \rho_{\mathrm{dom}}, \sigma, \mathbf{P}\rangle$$

назовём типом задачи о доминировании. Пусть

$$G_6 = \{g_{i,\cdot,m}(x_1,\ldots,x_n) =]x_i \cdot m[\colon i \in \{1,2,\ldots,n-1\},\ m = 1,2,3\ldots\}, \tag{4}$$

$$G_7 = \left\{ g_{i,<,a}(x_1,\ldots,x_n) = \begin{cases} 1, & \text{если } x_i < a, \\ 2, & \text{если } x_i \geqslant a\colon \end{cases} \quad i \in \{1,2,\ldots,n-1\},\ a \in (0,1] \right\}, \tag{5}$$

$$F_4 = \left\{ g_{n,\geqslant,a}(x_1,\ldots,x_n) = \begin{cases} 0, & \text{если } x_n < a, \\ 1, & \text{если } x_n \geqslant a\colon \end{cases} \quad a \in (0,1] \right\}, \tag{6}$$

$$\mathcal{F} = \langle F_4,\, G_6 \cup G_7\rangle. \tag{7}$$

Теорема 6 [17]. *Пусть вероятностная мера* $\mathbf{P}$ *определяется ограниченной функцией плотности распределения,* I *— задача информационного поиска типа* S_{dom}, $\mathcal{F}$ *— базовое множество, задаваемое соотношениями* (4)—(7). *Тогда если функция плотности вероятности* $p(x)$ *ограничена константой* c, *то*

$$0 < T\left(I, \mathcal{F}, \binom{k+n-1}{n} + (3+c)\cdot \sum_{i=1}^{n-1} \binom{k+i-1}{i}\right) - R(I) \leqslant 2n-1.$$

Этот результат был получен с помощью метода снижения размерности. Приведём краткое описание этого метода применительно к n-мерной задаче о доминировании. Возьмём произвольный запрос. Он описывает n требований к ответу: по каждой из n компонент элементы ответа не должны превышать соответствующую компоненту запроса. С помощью решения задачи о близости (опорной задачи, оптимальное решение которой приводится в теореме 3) мы получаем подмножество библиотеки, состоящее из всех записей, удовлетворяющих одному из n требований. Далее опять применяем к полученному подмножеству библиотеки задачу о близости и ещё раз снижаем размерность. Таким образом за $n-1$ применений задачи о близости (т. е. в среднем за $2(n-1)$ вычислений) мы придём к одномерной задаче о доминировании, оптимальное решение которой приводится в теореме 4.

Для двумерной задачи о доминировании исследовалось также решение задачи в фоновом режиме [25]. Для алгоритмов поиска в фоновом режиме предполагается наличие внешнего объекта, называемого пользователем. Элементы ответа на запрос при этом считаются поступающими по мере нахождения, каждый элемент ответа обрабатывается пользователем в течение некоторого времени, а сложность алгоритма определяется как время простоя пользователя. Найдено фоновое решение двумерной задачи о доминировании, которое в типичной ситуации при линейных затратах памяти имеет константную временну́ю сложность.

Задача интервального поиска состоит в поиске в конечном подмножестве n-мерного пространства всех тех точек, которые попадают в n-мерный параллелепипед-запрос. Пусть

$$X_{\text{int}\,n} = \{\tilde{x} = (u_1, v_1, \ldots, u_n, v_n)\colon 0 < u_i \leqslant v_i \leqslant 1,\ i = 1, 2, \ldots, n\}.$$

Отношение поиска $\rho_{\text{int}\,n}$ определено на $X_{\text{int}\,n} \times Y_{\text{int}\,n}$ и задаётся следующим соотношением:

$$(u_1, v_1, \ldots, u_n, v_n)\ \rho_{\text{int}\,n}\ (y_1, \ldots, y_n) \Longleftrightarrow u_i \leqslant y_i \leqslant v_i,\ i = 1, 2, \ldots, n.$$

Тогда тип

$$S_{\text{int}\,n} = \langle X_{\text{int}\,n}, Y_{\text{int}\,n}, \rho_{\text{int}\,n}, \sigma, \mathbf{P}\rangle$$

назовём типом интервального поиска. Пусть

$$G_8 = \{g^1_{i,\cdot,m}(u_1, v_1, \ldots, u_n, v_n) =]u_i \cdot m[\colon i \in \{1, 2, \ldots, n\},\ m = 1, 2, 3\ldots\}, \quad (8)$$

$$G_9 = \{g^2_{i,\cdot,m}(u_1, v_1, \ldots, u_n, v_n) =]v_i \cdot m[\colon i \in \{1, 2, \ldots, n-1\},\ m = 1, 2, 3\ldots\}, \quad (9)$$

$$G_{10} = \Bigg\{g^1_{i,\leqslant,a}(u_1, v_1, \ldots, u_n, v_n) =$$

$$= \begin{cases} 1, & \text{если}\ \ u_i \leqslant a, \\ 2, & \text{если}\ \ u_i > a\colon \end{cases} \quad i \in \{1, 2, \ldots, n\},\ a \in (0, 1]\Bigg\}, \quad (10)$$

$$G_{11} = \Bigg\{g^2_{i,<,a}(u_1, v_1, \ldots, u_n, v_n) =$$

$$= \begin{cases} 1, & \text{если}\ \ v_i < a, \\ 2, & \text{если}\ \ v_i \geqslant a\colon \end{cases} \quad i \in \{1, 2, \ldots, n-1\},\ a \in (0, 1]\Bigg\}, \quad (11)$$

$$G_{12} = \Bigg\{g_{-,m}(u_1, v_1, \ldots, u_n, v_n) =$$

$$= \begin{cases} 1, & \text{если}\ \ 0 \leqslant v_n - u_n < 1/m, \\ 2 & \text{в противном случае}\colon \end{cases} \quad m = 1, 2, 3\ldots\Bigg\}, \quad (12)$$

$$F_5 = \left\{f_a(u_1, v_1, \ldots, u_n, v_n) = \begin{cases} 1, & \text{если}\ \ u_n \leqslant a \text{ и } v_n \geqslant a, \\ 0 & \text{в противном случае}\colon \end{cases} \quad a \in (0, 1]\right\}. \quad (13)$$

$$\mathcal{F} = \langle F_5,\ G_8 \cup G_9 \cup G_{10} \cup G_{11} \cup G_{12}\rangle. \quad (14)$$

Теорема 7 [15,17]. *Пусть вероятностная мера* $\mathbf{P}$ *определяется ограниченной функцией плотности распределения с ограниченными частными производными первого порядка,* I *— задача информационного поиска типа* $S_{\text{int}\,n}$*,* $\mathcal{F}$ *— базовое множество, задаваемое соотношениями* (8)—(14)*. Тогда*

$$0 < T\left(I, \mathcal{F}, (4k + 2 + (1 + 6\,[\log_2 k]) \cdot c)\left(\frac{k(k+1)}{2}\right)^{n-1}\right) - R(I) \leqslant 4n + 1,$$

где c *— константа, зависящая от функции плотности распределения.*

Для равномерной вероятностной меры $c = 2$.

Этот результат был получен с использованием методов оптимальной декомпозиции и снижения размерности.

Приведём описание метода оптимальной декомпозиции применительно к одномерной задаче интервального поиска. Пусть нам дано множество $V = \{y_1, \ldots, y_k\}$, в котором мы должны производить поиск. Введём натуральное число m, являющееся параметром алгоритма. Если известна оценка сверху c функции плотности вероятности появления запросов (т. е. $p(x) \leqslant c$), то в качестве параметра m возьмём $m = 2c\lceil \log_2 k \rceil$, если же c неизвестна, то вместо неё можно взять любое число, например 2. Пусть $S = \{s_1, \ldots, s_m\}$, где $s_i = i/(m+1)$, $i = 1, 2, \ldots, m$. Произведём предобработку, заключающуюся в сортировке множества V в порядке возрастания и построении множества $L = \{l_1, \ldots, l_m\}$, где l_i — целое число, являющееся номером максимальной записи из V, не большей чем s_i, причём если такой записи не существует, то примем $l_i = 0$ ($i = 1, 2, \ldots, m$). Теперь поиск по произвольно взятому интервалу-запросу $x = (u, v)$ производится следующим образом.

Сначала вычисляется длина запроса x. Если она меньше, чем $1/m$, то в множестве V бинарным поиском находим ближайшую справа к точке u запись. Начиная с этой записи, просматриваем слева направо все записи из V и сравниваем с правым концом запроса — точкой v — до тех пор, пока очередная запись не станет больше v. В этом случае, помимо перечисления ответа, производится порядка $\log_2 k$ действий.

Если $v - u \geqslant 1/m$, то вычисляем номер $j = \rceil u \cdot m \lceil$ точки s_j, попадающей в интервал $[u, v]$. Начиная с записи с номером l_j, просматриваем справа налево записи из V и сравниваем с левым концом запроса — точкой u. Как только очередная запись окажется меньше u, мы, начиная с записи с номером $l_j + 1$, просматриваем слева направо записи из V и сравниваем с правым концом запроса — точкой v — до тех пор, пока очередная запись не станет больше v. В этом случае мы, помимо перечисления ответа, производим четыре лишних действия (сравниваем $v - u$ с $1/m$, вычисляем функцию $\rceil u \cdot m \lceil$, делаем одно лишнее действие, идя справа налево, и одно лишнее действие, идя слева направо).

Здесь множество L определяет точки разбиения на подзадачи, а каждая из подзадач является одномерной задачей о доминировании, которая, согласно теореме 4, решается очень просто.

Осталось заметить, что параметр m подобран так, что средняя сложность первого случая не превышает 1, если известна оценка сверху функции плотности вероятности, и не превышает некоторой константы, если эта оценка точно не известна. Это так, поскольку вероятность множества запросов, длина которых не больше $1/m$, не превышает $2c/m$.

Наконец, заметим, что данный алгоритм требует дополнительную память порядка $\log_2 k$, чтобы хранить множество L, в худшем случае время его работы равно $\log_2 k$ плюс время перечисления ответа, а в среднем — совсем небольшая константа (приблизительно 5) плюс перечисление ответа.

4. Влияние на оптимальное решение главных параметров модели

Как можно заметить, все рассмотренные задачи в некотором смысле хорошие, а именно все допускают снижение среднего времени поиска фактически до минимума. Возникает вопрос: насколько устойчиво свойство «хорошести», названное каноническим эффектом, при вариации параметров задач поиска? К параметрам, которые можно варьировать в задачах поиска, можно отнести следующие:

- базовое множество функций, характеризующее набор доступных средств;
- ограничения на объём информационного графа, характеризующий объём памяти соответствующего информационного графа алгоритма поиска;
- ε-расширение запроса; этот параметр позволяет получать, вообще говоря, новые типы задач поиска, он применим к классу задач, которые можно условно назвать непрерывными (к нему относятся задача о доминировании, задача интервального поиска и задача поиска идентичных объектов, когда пространство запросов, например, компактное подмножество вещественной прямой) и состоит в том, что запрос в новой задаче получается ε-расширением запроса старой задачи.

Как и следовало ожидать, сложность задачи поиска существенно зависит от выбора базового множества. Причём часто можно получить весь спектр, начиная от перебора (как самого сложного) и кончая алгоритмами, сложность которых практически совпадает с мощностной нижней оценкой. Проиллюстрируем этот тезис на примере одномерной задачи интервального поиска [14].

Теорема 8. *Если* $F_0 = \{\chi_a \colon a \in [0,1]\}$,

$$\chi_a(u,v) = \begin{cases} 1, & \text{если } \ u \leqslant a \ \text{и} \ v \geqslant a, \\ 0 & \text{в противном случае}, \end{cases}$$

$\mathcal{F}_0 = \langle F_0, \varnothing \rangle$, *то для произвольной задачи информационного поиска* $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1} \rangle$, *такой что все записи в библиотеке* V *различны, справедливо*

$$T(I, \mathcal{F}_0) = |V|.$$

Этот результат означает, что если базовое множество состоит только из характеристических функций записей, то перебор является оптимальным алгоритмом.

Теорема 9. *Если* $\mathcal{F}_1 = \langle F_1 \cup F_2, \varnothing \rangle$ *и функция плотности распределения вероятностей* $p(u,v)$, *определяющая меру* $\mathbf{P}$ *вероятностного пространства над множеством запросов* $X_{\mathrm{int}1}$, *ограничена, то для произвольной задачи информационного поиска* $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1} \rangle$ *выполнено*

$$T(I, \mathcal{F}_1) - R(I) \leqslant O\bigl(\sqrt{k}\bigr) \quad \text{при} \quad k \to \infty,$$

где $k = |V|$, причём существуют такая вероятностная мера $\mathbf{P}$ и такая задача информационного поиска $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1}\rangle$, где $|V| = k$, что

$$T(I, \mathcal{F}_1) - R(I) = O\big(\sqrt{k}\big) \quad \textit{при} \quad k \to \infty.$$

Теорема 10. *Если $\mathcal{F}_2 = \langle F_1 \cup F_2,\, G_3\rangle$, то для произвольной задачи информационного поиска $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1}\rangle$ выполняется*

$$T(I, \mathcal{F}_2) - R(I) \leqslant]\log_2 k[.$$

Теорема 11. *Если $\mathcal{F}_3 = \langle F_1 \cup F_2,\, G_2 \cup G_3\rangle$ и функция плотности распределения вероятностей $p(u, v)$, определяющая меру $\mathbf{P}$ вероятностного пространства над множеством запросов $X_{\mathrm{int}1}$, такая, что $p(u, v) \leqslant c = \mathrm{const}$, то для произвольной задачи информационного поиска $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1}\rangle$, в которой $|V| = k$, выполняется*

$$T(I, \mathcal{F}_3) - R(I) \leqslant]\log\log_2 k[+ 6 + 2c.$$

Теорема 12. *Если $\mathcal{F}_4 = \langle F_1 \cup F_2,\, G_1 \cup G_2 \cup G_3\rangle$ и функция плотности распределения вероятностей ограничена, то для произвольной задачи информационного поиска $I = \langle X_{\mathrm{int}1}, V, \rho_{\mathrm{int}1}\rangle$ выполняется*

$$T(I, \mathcal{F}_4) - R(I) \leqslant 5.$$

Зависимость сложности задачи поиска от объёма памяти более «плавная», чем от базового множества. В качестве примера этой зависимости можно рассмотреть случай, когда задача поиска есть задача поиска идентичных объектов [15].

Теорема 13. *Пусть $I = \langle X, V, \rho_{\mathrm{id}}\rangle$ — задача поиска идентичных объектов, $|V| = k$, $\mathcal{F}$ — базовое множество, задаваемое соотношениями* (1)—(3), *c — константа, ограничивающая функцию плотности распределения запросов,*

$$L_1(l) = \begin{cases} 0, & \textit{если} \quad l = 0, \\ ]\log_2 l[+ 1, & \textit{если} \quad l = 1, 2, 3, \\ \log_2 l + 2, & \textit{если} \quad l \geqslant 4, \end{cases}$$

функция, определённая на множестве целых неотрицательных чисел. Тогда

$$1 < T(I, \mathcal{F}, 2 \cdot k + m - 1) \leqslant$$
$$\leqslant \frac{c}{m}\left(\left(k - \left[\frac{k}{m}\right] \cdot m\right) \cdot L_1\left(\left[\frac{k}{m}\right] + 1\right) + \left(m - k + \left[\frac{k}{m}\right] \cdot m\right) \cdot L_1\left(\left[\frac{k}{m}\right]\right)\right) + 1.$$

В частности,

$$1 < T(I, \mathcal{F}, (2 + c) \cdot k) < 2$$

и $T(I, \mathcal{F}) \sim 1$ при $k \to \infty$.

При объёме памяти $2k$ мы имеем логарифмический поиск, а при увеличении объёма до $(2 + c)k$ мы плавно снижаем среднее время поиска до двух

операций. Эта зависимость более наглядна в асимптотической записи в случае равномерной вероятности запросов, т. е. когда $c = 1$:

$$T(I, \mathcal{F}, 2k + m) \lesssim 2 + \log_2 k - \log_2 m.$$

Эта формула «разумна» при $0 \leqslant m \leqslant k$. Таким образом, в данной ситуации выигрыш по времени логарифмически зависит от приращения объёма.

Если через k обозначить мощность библиотеки, то для двумерной задачи интервального поиска объём памяти, необходимый алгоритму, на котором достигается оценка теоремы 7, равен $O(k^3)$. С целью понижения объёма памяти в [23] разработана модификация алгоритма Бентли—Маурера, сохраняющая порядки времени поиска в худшем случае и объёма памяти при снижении среднего времени поиска (без времени перечисления ответа) до константы. На основе этого алгоритма получена следующая оценка.

Теорема 14. *Пусть I — двумерная задача интервального поиска, вероятностная мера $\mathbf{P}$ определяется ограниченной функцией плотности распределения с ограниченными частными производными первого порядка, $\mathcal{F}_3$ — базовое множество, задаваемое соотношениями* (8)—(14)*. Тогда для любого натурального M, такого что $1 \leqslant M \leqslant 2 \ln k$, выполняется*

$$0 \leqslant T\left(I, \mathcal{F}_3, \frac{2}{3} M k^{1+2/M} + O\big(k^{1+1/M}\big)\right) - R(I) \leqslant 14M - 4.$$

Таким образом, здесь также наблюдается «плавная» зависимость сложности задачи поиска от объёма памяти.

Ситуация, возникающая при обобщении задач за счёт ε-расширения запроса, не однозначна. Так, в задаче о доминировании и задаче интервального поиска при малых ε результаты, описанные в теоремах 6 и 7, полностью сохраняются, так как ε-расширение приводит лишь к вымыванию «малых» запросов, а поскольку их доля мала, то это не отражается на результате. В случае задачи поиска идентичных объектов в геометрической интерпретации, когда множество запросов — отрезок $[0, 1]$ вещественной прямой, картина более интересная. При малых ε (например, при $\varepsilon < 1/k^2$, где k — мощность библиотеки) справедлива ситуация, описанная в теореме 13. При больших ε задача превращается в упрощённую версию одномерной задачи интервального поиска и результат будет аналогичен результату, описанному в теореме 12.

5. Заключение

Информационно-графовая модель позволяет предложить новую технологию проектирования физической организации баз данных. По этой технологии на начальном этапе выделяются классы однотипных вопросов к базе данных, оформляемые в виде типов задач поиска. Множество баз данных задаёт конкретную задачу поиска данного типа. Для каждой задачи поиска выделяется множество

элементарных операций над запросами, оформляемое в виде базового множества, и решается задача синтеза оптимального информационного графа, решающего данную задачу поиска. Полученный информационный граф описывает оптимальную структуру данных, соответствующую заданным целям оптимизации (среднему времени поиска, времени поиска в худшем случае, объёму памяти).

Тем самым один информационный граф описывает структурную часть базы данных, обрабатывающую один класс однотипных вопросов к базе данных, а сама база данных в информационно-графовой модели представляется как совокупность нескольких информационных графов, охватывающих весь спектр вопросов к базе данных.

Поскольку в работе рассматриваются наиболее распространённые типы задач поиска, решение проблемы оптимального синтеза для данных задач позволяет для большинства случаев, возникающих при проектировании физической организации баз данных, иметь готовые рекомендации.

Среди основных направлений развития данной теории можно выделить следующие.

Представляется естественным дальнейшее исследование как перечисленных выше типов задач поиска, таких, например, как задачи поиска идентичных объектов [24], включающего [21]) и интервального [19, 23] поиска, так и новых типов задач поиска. Так, в [8, 9] исследовалось решение задачи о метрической близости на булевом кубе как в рамках информационно-графовой модели, так и в классе автоматных схем специального вида, приближенных по своим характеристикам к реальным чипам (интегральным схемам). В цикле работ [2—7] было проведено исследование задачи интервального поиска на булевом кубе. Для этой задачи получены асимптотическое поведение среднего времени поиска в классе сбалансированных древовидных схем и оценки функции Шеннона сложности в классе древовидных схем, предложены алгоритмы построения информационных деревьев и получены оценки их сложности. Особо хотелось бы выделить направление, связанное с исследованием функциональной сложности информационных графов [19, 23, 63], т. е. функции зависимости времени поиска от объёма доступной памяти. Эти исследования очень полезны для практики, так как позволяют в зависимости от имеющихся ресурсов памяти подбирать наиболее быстрые алгоритмы поиска.

Перспективными представляются направления, связанные с исследованием параллельных, фоновых и нечётких задач поиска. В [20] понятие информационного графа распространено на случай параллельных алгоритмов поиска и предложено оптимальное параллельное решение одномерной задачи о доминировании. В [25] предложено обобщение информационных графов для фоновых алгоритмов поиска и разработан хороший фоновый алгоритм поиска для двумерной задачи о доминировании. Окончательное фоновое решение двумерной задачи о доминировании получено Д. В. Ефремовым с помощью исследования тонких свойств частично-упорядоченных множеств, задаваемых перестановками. В [50, 51] предложена математическая модель нечёткого информационного

поиска и исследованы методы перехода от чётких задач поиска к нечётким и наоборот.

Интересным представляется исследование динамических баз данных, представленное в [33,34]. Идейно к этим работам примыкает работа [49], связанная с базами данных движущихся объектов.

В [31,32] базы данных рассматриваются как множество независимых одинаково распределённых случайных величин. Это позволяет исследовать средние характеристики алгоритмов поиска при вариации баз данных.

Среди задач поиска в базах данных, кроме задач на перечисление ответа, которые описывались выше, есть ещё задачи на поиск представителя, когда достаточно найти один объект, удовлетворяющий запросу. Такие задачи интересны как сами по себе, так и как вспомогательный аппарат в фоновых задачах поиска. В [45] предложен близкий к оптимальному алгоритм поиска представителя для задачи о метрической близости.

В [53,54] исследуется задача реализации булевых функций информационными графами. Эта задача близка к задаче реализации булевых функций контактными схемами, но если сложность контактных схем характеризует схемную сложность функций, то сложность информационного графа, реализующего булеву функцию, отражает сложность вычисления значения функции на входном наборе.

Следующие два направления не связаны напрямую с информационно-графовой моделью, но относятся к тематике баз данных.

Первое направление связано с протоколами доступа к данным без раскрытия запросов (PIR-протоколами), которые позволяют сохранить в секрете интересы пользователей от владельцев баз данных. В серии работ [37—40] проведено исследование коммуникационной сложности таких протоколов, под которой понимается количество бит, которыми обмениваются пользователь и владельцы базы данных в ходе протокола. Получен критерий принадлежности PIR-протоколов к классу невырожденных (вырожденным называется PIR-протокол с коммуникационной сложностью, равной длине базы данных), построены оптимальные по порядку PIR-протоколы для широкого класса параметров, исследована степень раскрытия PIR-протоколов.

Второе направление связано с проблемами поиска информации в Интернете. Идейно оно близко к таким направлениям, как компьютерное обучение, распознавание образов, теория тестов. В математической постановке эта задача сводится к задаче расшифровки функций из специальных классов. Этому направлению посвящены работы [10,43,44].

Литература

[1] Андреев А. Е. Метод бесповторной редукции синтеза самокорректирующихся схем // ДАН СССР. — 1985. — Т. 283, № 2. — С. 265—269.

[2] Блайвас Т. Д. Оптимальное решение задачи интервального поиска на булевом кубе в классе сбалансированных древовидных схем // Интеллект. сист. — 2002-2003. — Т. 7, № 1-4. — С. 223—245.

[3] Блайвас Т. Д. Асимптотика сложности интервального поиска на булевом кубе в классе сбалансированных деревьев // Дискрет. мат. — 2004. — Т. 16, № 4. — С. 65—78.

[4] Блайвас Т. Д. Один алгоритм решения задачи интервального поиска на булевом кубе // Интеллект. сист. — 2004. — Т. 8, № 1-4. — С. 389—408.

[5] Блайвас Т. Д. О сложности интервального поиска на булевом кубе: Дис.... канд. физ.-мат. наук. — М., 2005.

[6] Блайвас Т. Д. Сложность поиска по маске для алгоритма с жёстким порядком проверок // Интеллект. сист. — 2005. — Т. 9, № 1-4. — С. 347—362.

[7] Блайвас Т. Д. Функция Шеннона сложности интервального поиска на булевом кубе в классе деревьев // Дискрет. мат. — 2006. — Т. 18, № 2. — С. 111—122.

[8] Быченкова Е. С. Асимптотическое решение задачи о метрической близости для одного базового множества функций // Интеллект. сист. — 2001. — Т. 6, № 1-4. — С. 221—230.

[9] Быченкова Е. С. Оптимальный по порядку метод синтеза одного поискового оператора в классе автоматных схем специального вида // Дискрет. мат. — 2003. — Т. 15, № 1. — С. 132—156.

[10] Воронин Б. В., Осокин В. В. О сложности расшифровки существенных переменных функции, задающей разбиение булевого куба // Интеллект. сист. — В печати.

[11] Гасанов Э. Э. Оптимальные информационные сети для отношений поиска, являющихся отношениями линейного квазипорядка // Конструкции в алгебре и логике. — Тверь: Изд-во Тверского гос. ун-та, 1990. — С. 11—17.

[12] Гасанов Э. Э. Об одной математической модели информационного поиска // Дискрет. мат. — 1991. — Т. 3, № 2. — С. 69—76.

[13] Гасанов Э. Э. Нижняя оценка сложности информационных сетей для одного класса задач информационного поиска // Дискрет. мат. — 1992. — Т. 4, № 3. — С. 118—127.

[14] Гасанов Э. Э. Об одномерной задаче интервального поиска // Дискрет. мат. — 1995. — Т. 7, № 2. — С. 40—60.

[15] Гасанов Э. Э. Мгновенно решаемые задачи поиска // Дискрет. мат. — 1996. — Т. 8, № 3. — С. 119—134.

[16] Гасанов Э. Э. Нижняя оценка сложности информационных сетей для одного отношения частичного порядка // Дискрет. мат. — 1996. — Т. 8, № 4. — С. 108—122.

[17] Гасанов Э. Э. Функционально-сетевые базы данных и сверхбыстрые алгоритмы поиска. — М.: Изд. центр РГГУ, 1997.

[18] Гасанов Э. Э. Теория сложности информационного поиска. — М.: Изд-во механико-математического ф-та МГУ, 2005.

[19] Гасанов Э. Э., Ерохин А. Н. Линейный по памяти неперебoрный алгоритм решения двумерной задачи интервального поиска // Дискрет. мат. — 2004. — Т. 16, № 4. — С. 49—64.

[20] Гасанов Э. Э., Ерохина Е. Р. Моделирование и сложность поиска в многопроцессорных системах // Дискрет. мат. — 1999. — Т. 11, № 3. — С. 63—82.

[21] Гасанов Э. Э., Косолапов А. В. К вопросу о древовидности оптимальных информационных сетей включающего поиска // Интеллект. сист. — 1998. — Т. 3, № 1-2. — С. 167—192.

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

[23] Гасанов Э. Э., Кузнецова И. В. О функциональной сложности двумерной задачи интервального поиска // Дискрет. мат. — 2002. — Т. 14, № 1. — С. 114—141.

[24] Гасанов Э. Э., Луговская Ю. П. Константный в худшем случае алгоритм поиска идентичных объектов // Дискрет. мат. — 1999. — Т. 11, № 4. — С. 139—144.

[25] Гасанов Э. Э., Мхитарова Т. В. Об одной математической модели фоновых алгоритмов поиска и быстрый фоновый алгоритм двумерной задачи о доминировании // Фундамент. и прикл. мат. — 1997. — Т. 3, вып. 3. — С. 759—773.

[26] Ершов А. П. О программировании арифметических операторов // ДАН СССР. — 1958. — Т. 118. — С. 427—430.

[27] Ефремов Д. В. Свойства частично-упорядоченных множеств, задаваемых перестановками // Интеллект. сист. — 2005. — Т. 9, № 1-4. — С. 363—380.

[28] Кнут Д. Искусство программирования для ЭВМ. — М.: Мир, 1978. — Т. 3: Сортировка и поиск.

[29] Кудрявцев В. Б. Функциональные системы. — М.: Изд-во Моск. ун-та, 1982.

[30] Кудрявцев В. Б., Гасанов Э. Э., Подколзин А. С. Введение в теорию интеллектуальных систем. — М.: Изд-во ф-та ВМиК МГУ, 2006.

[31] Кучеренко Н. С. Сложность поиска идентичных объектов для случайных баз данных // Интеллект. сист. — 2007. — Т. 11, № 1-4. — С. 495—516.

[32] Кучеренко Н. С. Средняя сложность поиска идентичных объектов для случайных неравномерных баз данных // Дискрет. мат. — В печати.

[33] Лапшов И. С. Динамические базы данных, основывающиеся на хешировании методом цепочек // Интеллект. сист. — 2005. — Т. 9, № 1-4. — С. 191—207.

[34] Лапшов И. С. Динамические базы данных с оптимальной по порядку временной сложностью // Дискрет. мат. — 2008. — Т. 20, № 3. — С. 89—100.

[35] Ли Д., Препарата Ф. Вычислительная геометрия. Обзор // Кибернет. сб. — 1987. — Вып. 24. — С. 5—96.

[36] Лупанов О. Б. О синтезе некоторых классов управляющих систем // Проблемы кибернетики. Вып. 10. — М.: Наука, 1963. — С. 63—97.

[37] Майлыбаева Г. А. Границы вырожденности протоколов доступа к данным без раскрытия запроса // Дискрет. мат. — 2006. — Т. 18, № 2. — С. 98—110.

[38] Майлыбаева Г. А. Коммуникационная сложность протоколов доступа к данным без раскрытия запросов: Дис.... канд. физ.-мат. наук. — М., 2007.

[39] Майлыбаева Г. А. Точное значение коммуникационной сложности для одного класса PIR-протоколов // Интеллект. сист. — 2007. — Т. 11, № 1-4. — С. 167—200.

[40] Майлыбаева Г. А. Порядок коммуникационной сложности для одного класса PIR-протоколов // Дискрет. мат. — 2008. — Т. 20, № 3. — С. 136—146.

[41] Мартин Дж. Организация баз данных в вычислительных системах. — М.: Мир, 1980.

[42] Ньюмен У. М., Спруэлл Р. Ф. Основы интерактивной машинной графики. — М.: Мир, 1976.

[43] Осокин В. В. Асимптотически оптимальный алгоритм расшифровки разбиения булевого куба на подкубы // Интеллект. сист. — 2007. — Т. 11, № 1-4. — С. 587—606.

[44] Осокин В. В. О сложности расшифровки разбиения булевого куба на подкубы // Дискрет. мат. — 2008. — Т. 20, № 2. — С. 46—62.

[45] Пивоваров А. П. Поиск представителя в задаче о метрической близости // Интеллект. сист. — В печати.

[46] Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. — М.: Мир, 1989.

[47] Решетников В. Н. Алгебраическая теория информационного поиска // Программирование. — 1979. — № 3. — С. 68—74.

[48] Селтон Г. Автоматическая обработка, хранение и поиск информации // М.: Советское радио, 1973.

[49] Скиба Е. А. Логарифмическое решение задачи об опасной близости // Интеллект. сист. — 2007. — Т. 11, № 1-4. — С. 645—676.

[50] Фещук А. А. К вопросу анализа нечетких информационных графов // Дискрет. мат. — 2002. — Т. 14, № 2. — С. 65—84.

[51] Харина А. А. О сведении нечёткого информационного поиска к информационному поиску большей размерности // Интеллект. сист. — 2005. — Т. 9, № 1-4. — С. 57—76.

[52] Шеннон К. Работы по теории информации и кибернетике. — М.: Изд. иностр. лит., 1963.

[53] Шуткин Ю. С. Синтез информационных графов для предполных классов булевых функций // Интеллект. сист. — 2007. — Т. 11, № 1-4. — С. 689—604.

[54] Шуткин Ю. С. О реализации булевых функций информационными графами // Дискрет. мат. — В печати.

[55] Ben-Or M. Lower bounds for algebraic computation trees // Proc. 15th ACM Ann. Symp. Theory Comput. (April 1983). — P. 80—86.

[56] Bentley J. L., Friedman J. H. Data structures for range searching // Comput. Surveys. — 1979. — Vol. 11. — P. 397—409.

[57] Bentley J. L., Maurer H. A. Efficient worst-case data structures for range searching // Acta Inform. — 1980. — Vol. 13. — P. 155—168.

[58] Bentley J. L., Stanat D. F. Analysis of range searching in quad trees // Inform. Process. Lett. — 1975. — Vol. 3. — P. 170—173.

[59] Bolour A. Optimal retrieval algorithms for small region queries // SIAM J. Comput. — 1981. — Vol. 10. — P. 721—741.

[60] Chazelle B. M. Filtering search: A new approach to query-answering // Proc. 24th IEEE Ann. Symp. Found. Comput. Sci. (November 1983). — C. 122—132.

[61] Fredman M. L., Baskett F., Shustek J. An algorithm for finding nearest neighbors // IEEE Trans. Comput. — 1975. — Vol. C-24. — P. 1000—1006.

[62] Fredman M. L., Bentley J. L., Finkel R. A. An algorithm for finding best match in logarithmic expected time // ACM Trans. Math. Software. — 1977. — Vol. 3, no. 3. — P. 209—226.

[63] Gasanov E. E. On functional complexity of two-dimensional Manhattan metrics closeness problem // Emerging Database Research in East Europe. Proc. of the Pre-Conference Workshop of VLDB 2003. — Berlin, 2003. — P. 51—56.

[64] Lee D. T., Wong C. K. Worst case analysis for region and partial region searches in multidimensional binary search trees and balanced quad trees // Acta Inform. — 1977. — Vol. 9. — P. 23—29.

[65] Lueker G. S. A data structure for ortogonal range queries // Proc. 19th IEEE Ann. Symp. Found. Comput. Sci. (1978). — P. 28—34.