Сложность поиска по маске для алгоритма с жестким порядком проверок

Т. Д. Блайвас

Для задачи поиска по маске на булевом кубе предложен алгоритм построения решающей древовидной схемы по упорядоченной библиотеке и заданному порядку проверок координат. Показано, что почти все построенные схемы имеют одинаковую по порядку временну́ю сложность, меньшую сложности в классе сбалансированных древовидных схем [2].

1. Введение

В работе исследуется следующая задача информационного поиска. Имеется некоторое подмножество V n-мерного булева куба B_2^n, называемое библиотекой. По произвольной маске $x = (x_1, \ldots, x_n) \in \{0, 1, 2\}^n$ требуется определить все такие элементы $y \in V$, $y = (y_1, \ldots, y_n)$, называемые записями, для каждой координаты которых выполнено: либо $x_i = 2$, либо $y_i = x_i$.

Приведем интерпретацию данной задачи. Допустим, некто имеет название ценной для него книги, но некоторые буквы в названии от времени стерлись. В городской библиотеке требуется определить, есть ли книги с подходящим названием, и, если есть, выдать их читателю.

Задачу можно решать, если на каждом шаге алгоритма проверять условие $(y_j = x_j) \vee (x_j = 2)$, $j \in \{1, \ldots, n\}$ для фиксированного набора компонент записи.

Предложен алгоритм построения древовидной решающей схемы, на вход которого поступает библиотека, порядок ее элементов и по-

рядок проверок координат записей. Показано, что алгоритм не зависит от порядка поступления записей библиотеки. Показано, что для k-элементных библиотек ($k = \bar{o}(2^n)$) из n-мерного ($n = \bar{o}(2^k)$) булева куба доля решающих деревьев, чья сложность по порядку равна $(k/\log_2 k)^{\log_2 4/3}$, асимтотически равна 1 при $n \to \infty$, $k \to \infty$.

2. Основные понятия и формулировка результатов

Мы будем использовать терминологию и обозначения из работы [4], но поскольку в данной работе рассматриваются только древовидные схемы, то здесь будет приведена несколько упрощенная версия понятия информационного графа.

Если X — множество символов запросов с заданным на нем вероятностным пространством $\langle X, \sigma, \mathbf{P}\rangle$, где σ — алгебра подмножеств множества X, $\mathbf{P}$ — вероятностная мера на σ; Y — множество символов данных (записей); ρ — бинарное отношение на $X \times Y$, называемое отношением поиска; то пятерка $S = \langle X, Y, \rho, \sigma, \mathbf{P}\rangle$ называется *типом*. Тройка $I = \langle X, V, \rho\rangle$, где V — некоторое конечное подмножество множества Y, называемое библиотекой, называется задачей информационного поиска (ЗИП) типа S. Содержательно ЗИП $I = \langle X, V, \rho\rangle$ состоит в перечислении для произвольно взятого запроса $x \in X$ всех тех и точно тех записей $y \in V$, что $x\rho y$. Если $\mathcal{F}$ — суть множества символов одноместных предикатов, определенных на X, $\mathcal{F}$ называется базовым множеством и описывает множество элементарных операций, используемых при решении задачи информационного поиска.

Над базовым множеством $\mathcal{F}$ определяется понятие *информационного графа* (ИГ). В конечной многополюсной ориентированной сети выбирается вершина — полюс, называемая корнем. Остальные полюса называются листьями и им приписываются записи из Y. Ребрам ИГ приписываются предикаты из множества $\mathcal{F}$. Таким образом нагруженную многополюсную ориентированную сеть называют информационным графом над базовым множеством $\mathcal{F}$. Затем определяется *функционирование ИГ*. Предикатное ребро проводит запрос $x \in X$, если предикат ребра истинен на x; ориентированная цепочка ребер

проводит x, если каждое ребро цепочки проводит x; запрос x проходит в вершину β ИГ, если существует ориентированная цепь, ведущая из корня в вершину β, которая проводит x; запись y, приписанная листу α, попадает в ответ ИГ на x, если x проходит в лист α. Ответом ИГ U на запрос x называют множество записей, попавших в ответ U на x, и обозначают его $\mathcal{J}_U(x)$. Эту функцию $\mathcal{J}_U(x)$ считают результатом функционирования ИГ U.

ИГ U *разрешает* ЗИП $I = \langle X, V, \rho\rangle$, если $\mathcal{J}_U(x) = \{y \in V : x\rho y\}$.

Вводится *сложность ИГ*. Предикат $\varphi_\beta(x)$ истинный на x, если x проходит в вершину β, и ложный в противном случае, называется функцией фильтра вершины β. *Сложностью ИГ* U *на запросе* $x \in X$ называется число $T(U,x) = \sum_{\beta\in\mathcal{R}} \psi_\beta \cdot \varphi_\beta(x)$, где $\mathcal{R}$ — множество вершин ИГ U, ψ_β — количество ребер, исходящих из вершины β. Эта величина равна числу функций, вычисленных алгоритмом поиска, определяемым ИГ U, на запросе x.

Если каждая функция из $\mathcal{F}$ измерима (относительно алгебры σ), то для любого ИГ U над $\mathcal{F}$ функция $T(U,x)$ измерима.

Сложностью ИГ U называется математическое ожидание величины $T(U,x)$, равное $T(U) = \mathbf{M}_x\, T(U,x)$. Она характеризует среднее время поиска.

Легко показать, что

$$T(U) = \sum_{\beta\in U} \psi_\beta \cdot P(N_{\varphi_\beta}(x)). \tag{1}$$

Если f — предикат на множестве X, то $N_f(x) = \{x \in X : f(x) = 1\}$. Сложностью ребра, исходящего из вершины β назовем число $P(N_{\varphi_\beta}(x))$. Согласно (1), сложность ИГ равна сумме сложностей ребер. Обозначим через $T_i(D)$, i — натуральное число, сложность первых i ярусов ребер дерева D.

Рассмотрим следующую ЗИП. Имеется некоторое k-элементное подмножество n-мерного булева куба $V \in B_2^n$ (библиотека). На булевом кубе задан некоторый интервал (u,w), где $u = (u_1,\ldots,u_n)$, $w = (w_1,\ldots,w_n)$, и $u \preceq w$, то есть $u_i \leqslant w_i$, $\forall i = 1,\ldots,n$. Требуется определить все элементы $y \in V$, удовлетворяющие условию $u \preceq y \preceq w$.

Очевидно, что если $u_i = 1$ для некоторого i, то и $w_i = 1$, а, следовательно, и $y_i = 1$. Аналогично, если $w_i = 0$, то и $y_i = 0$. Таким образом, вышеописанная ЗИП сводится к следующей: есть библиотека $V \in B_2^n$, $|V| = k$, берем запрос $x = (x_1, \ldots, x_n)$ — трехзначный вектор, компоненты которого могут быть равны либо 1, либо 0, либо 2: если $u_i = 1$, то $x_i = 1$, если $w_i = 0$, то $x_i = 0$, иначе $x_i = 2$. Для данного запроса $x = (x_1, \ldots, x_n)$ хотим найти все $y = (y_1, \ldots, y_n) \in V$, для которых $y_i = x_i$, если $x_i = 1$ или $x_i = 0$, и y_i любое из $\{0, 1\}$, если $x_i = 2$.

Получаем тип задач $S_n = \langle B_3^n, B_2^n, \rho, \sigma, P \rangle$, где B_3^n и B_2^n — трехзначный и двузначный (булев) кубы, соответственно, ρ : $x \rho y \Leftrightarrow (x_i = y_i) \vee (x_i = 2)$, σ — множество подмножеств B_3^n, и P — равномерная вероятностная мера на B_3^n, то есть для любого $x \in B_3^n$ $P(x) = 3^{-n}$.

Вершину со степенью полуисхода, равной нулю назовем *висячей вершиной. Информационным деревом (ИД)* назовем ИГ без циклов, множество листьев которого совпадает с множеством висячих вершин, и все ребра которого ориентированы от корня к листьям. *Высотой ИД* назовем длину максимального пути из корня в лист.

Пусть $x \in \{0, 1, 2\}$ и $y \in \{0, 1, 2\}$. Определим функцию x^y:

$$x^y = \begin{cases} x, & \text{если} \quad (y = 1) \& (x \neq 2) \\ \bar{x}, & \text{если} \quad (y = 0) \& (x \neq 2) \\ 1, & \text{если} \quad (y = 2) \vee (x = 2) \end{cases},$$

причем $\bar{x}$ понимается здесь как булевое отрицание.

Определим понятие яруса вершин. Вершиной первого яруса будем называть корень. Для любого числа i, не большего высоты дерева плюс 1, вершинами i-го яруса назовем такие вершины, из которых длина пути в корень равна $i-1$. Будем говорить, что ребро находится на ярусе с номером i, если оно исходит из вершины яруса с номером i.

Множество задач $I = \langle B_3^n, V, \rho \rangle$ типа S_n, где $|V| = k$, обозначим через $\mathcal{I}(n, k)$.

Пусть ребра информационного дерева занумерованы некоторым образом. *Номером цепи ребер* будем называть номер первого ребра цепи. *Нагрузкой цепи ребер* будем называть конъюнкцию нагрузок всех ребер цепи. *Полной цепью* из вершины v будем называть цепь

ребер, конечная вершина которой имеет полустепень исхода, не равную 1, а все остальные вершины цепи имеют полустепень исхода, в точности равную 1.

Будем рассматривать следующие функции и пользоваться следующими обозначениями и операциями.

Обозначим через $C(v,p,m)$ следующую операцию: из вершины v с двоичным кодом $(a_1,\dots,a_m)$ строим ориентированную от v цепочку ребер длины $n-m$, всем вершинам новой цепочки последовательно (по ориентации) приписываем двоичный код $(a_1,\dots,a_m,y^p_{j_{m+1}},\dots,y^p_{j_{m+l}})$, $l=1,\dots,n-m$, длины кодов вершин возрастают от $m+1$ до n, а всем ребрам цепочки последовательно (по ориентации) приписываем функции $x^{y^p_{j_{m+l}}}_{j_{m+l}}$, $l=1,\dots,n-m$. Листу последнего ребра приписываем запись y^p.

Для задачи $I=\langle B_2^n,V,\rho\rangle\in\mathcal{I}(n,k)$, перестановки элементов библиотеки $\sigma_0=(i_1,\dots,i_k)\in S_k$ и перестановки номеров переменных $\sigma=(j_1,\dots,j_n)$ решающее дерево будем строить при помощи следующего алгоритма $A(V,\sigma_0,\sigma)$.

Алгоритм $A(V,\sigma_0,\sigma)$**:**
В множество вершин дерева D кладем корень с кодом $v=*$ (пустой символ).
$C(*,i_1,0)$
$v=*$
Для каждого i_p, $p=2,\dots,k$ выполняем
Шаг 1. $N=1$
Шаг 2. Если есть ребро, исходящее из v, с нагрузкой $x^{y^{i_p}_{j_N}}_{j_N}$,
то переходим в вершину с кодом $v=(v,y^{i_p}_{j_N})$.
$N=N+1$
Переходим к шагу 2.
Шаг 3. Иначе $C(v,i_p,N)$.

Таким образом получаем дерево $D=A(V,\sigma_0,\sigma)$.

Утверждение 1. *Для любой библиотеки $V\subseteq B_2^n$, для любой перестановки $\sigma\in S_n$, для любых $\sigma_1\in S_k$ и $\sigma_2\in S_k$*

$$A(V,\sigma_1,\sigma)=A(V,\sigma_2,\sigma)=A(V,\sigma).$$

Доказательство. Пусть $D = A(V, \sigma_0, \sigma)$ для некоторой перестановки $\sigma_0 \in S_k$ и перестановки $\sigma = (j_1, \ldots, j_n)$. По построению, D содержит вершину с двоичным кодом $(v_1, \ldots, v_p)$, $p \in \{1, \ldots, n\}$, тогда и только тогда, когда существует $y = (y_1, \ldots, y_n) \in V$, такой что $y_{j_m} = v_m$, $m = 1, \ldots, p$. Значит, множество вершин дерева D не зависит от σ_0. По построению дерева D, смежными являются только те вершины, длина кодов которых отличается на единицу, и меньший код является префиксом большего. Если $(v_1, \ldots, v_p)$ и $(v_1, \ldots, v_p, v_{p+1})$ — коды смежных вершин, то нагрузка инцидентного им ребра совпадает с $x_{j_{p+1}}^{v_{p+1}}$, ребро направлено в сторону кода большей длины. Записи в дереве приписаны тем вершинам, с кодом которых они совпадают. Таким образом, результат алгоритма не зависит от перестановки σ_0.

Утверждение 1 доказано.

Положим $\mathcal{V}(n, k) = \{V \in B_2^n, |V| = k\}$.

Пусть $V \in \mathcal{V}(n, k)$. Положим $\bar{T}1(V) = M_\sigma T(A(V, \sigma))$, где $\sigma \in S_n$. По определению математического ожидания,

$$\bar{T}1(V) = \frac{\sum\limits_{\sigma \in S_n} T(A(V, \sigma))}{n!}.$$

Положим $\bar{T}1(n, k) = M_V \bar{T}1(V)$, где $V \in B_2^n : |V| = k$. Поскольку $|\mathcal{V}(n, k)| = C_{2^n}^k$,

$$\bar{T}1(n, k) = \frac{\sum\limits_{V \in \mathcal{V}(n,k)} \bar{T}1(V)}{C_{2^n}^k} = \\ = \frac{\sum\limits_{V \in \mathcal{V}(n,k)} \sum\limits_{\sigma \in S_n} T(A(V, \sigma))}{n! C_{2^n}^k} = \frac{\sum\limits_{\sigma \in S_n} \sum\limits_{V \in \mathcal{V}(n,k)} T(A(V, \sigma))}{n! C_{2^n}^k}.$$

Теорема 1. *При $n \to \infty$, $k \to \infty$, $n = \bar{o}(2^k)$, $n \geqslant 4$, $k = \bar{o}(2^n)$*

$$\bar{T}(n, k) \asymp \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}.$$

2.1. Вспомогательные утверждения

Пусть $\sigma = (1, 2, \ldots, n)$ — тождественная перестановка. Обозначим через E_i^V число ребер на ярусе с номером i в дереве $A(V, \sigma)$, $i = 1, \ldots, n$. Положим $E_i = M_V E_i^V = \frac{\sum_{V \in B_2^n : |V| = k} E_i^V}{C_{2^n}^k}$.

Лемма 1.

$$\bar{T}1(n, k) = \sum_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i.$$

Доказательство. Пусть $\mathcal{D}(n, k)$ — класс усеченных бинарных информационных деревьев с k висячими вершинами, решающих задачи $I \in \mathcal{I}(n, k)$ над базисом $\mathcal{F}_1$ переменных. *Структурой* $S(D)$ *информационного дерева* $D \in \mathcal{D}(n, k)$ назовем ориентированный граф, порождающий D.

Так как вероятность запроса пройти в некоторую вершину v информационного дерева D над базисом $\mathcal{F}_1$ зависит только от суммарной длины конъюнкции нагрузок, приписанных от корня до ребра с конечной вершиной v, то, согласно 1, сложность дерева над каждым из базисов зависит только от его структуры. Обозначим через $\mathcal{S}(n, k)$ множество структур деревьев из $\mathcal{D}(n, k)$. Вершиной нулевого яруса назовем корень. Вершину v', смежную с вершиной v $(i-1)$-го яруса, назовем вершиной i-го яруса, если ребро, инцидентное v и v' направлено от v к v', $i = 1, \ldots, n$.

Сложностью структуры $S \in \mathcal{S}(n, k)$ назовем величину

$$T(S) = \sum_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} \cdot E_i(S),$$

где $E_i(S)$ — суммарное число ребер, выходящих из вершин $(i-1)$-го яруса структуры S. Очевидно, $T(S(D)) = T(D)$ для любого дерева $D \in \mathcal{D}(n, k)$ со структурой $S(D) \in \mathcal{S}$.

Заметим, что каждой паре (V, σ), $V \in B_2^n$, $|V| = k$, $\sigma \in S_n$, однозначно соответствует (определяется алгоритмом A) некоторое усеченное бинарное дерево со структурой $S \in \mathcal{S}(n, k)$, а каждой паре (S, σ), $S \in \mathcal{S}(n, k)$, $\sigma \in S_n$, однозначно соответствует $V \in B_2^n$, $|V| = k$.

То есть, если фиксировать перестановку $\sigma \in S_n$, устанавливается взаимно однозначное соответствие между множествами $\mathcal{S}(n,k)$ и $\mathcal{V}(n,k)$. Так как $T(D) = T(S(D))$, то для фиксированной перестановки σ

$$\sum_{V \in \mathcal{V}(n,k)} T(A(V,\sigma)) = \sum_{S \in \mathcal{S}(n,k)} T(S),$$

и данная сумма не зависит от перестановки. Возьмем $\sigma = (1, 2, \ldots, n)$.

$$\bar{T}1(n,k) = \frac{\sum\limits_{\sigma \in S_n} \sum\limits_{V \in \mathcal{V}(n,k)} T(A(V,\sigma))}{n! C_{2^n}^k} = \frac{\sum\limits_{\sigma \in S_n} \sum\limits_{S \in \mathcal{S}(n,k)} T(S)}{n! C_{2^n}^k} =$$
$$= \frac{n! \sum\limits_{V \in \mathcal{V}(n,k)} T(A(V,\sigma))}{n! C_{2^n}^k} = \frac{\sum\limits_{V \in \mathcal{V}(n,k)} T(A(V,\sigma))}{C_{2^n}^k}.$$

Везде далее $\sigma = (1, 2, \ldots, n)$ — тождественная перестановка. Тогда получаем

$$T(A(V,\sigma)) = \sum_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i^V,$$

$$\bar{T}1(n,k) = \frac{\sum\limits_{V \in \mathcal{V}(n,k)} \sum\limits_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i^V}{C_{2^n}^k} =$$
$$= \frac{\sum\limits_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} \cdot \left(\sum\limits_{V \in \mathcal{V}(n,k)} E_i^V \right)}{C_{2^n}^k} = \sum_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i.$$

Лемма доказана.

Для доказательства теоремы нам понадобятся следующие факты.

Лемма 2. *Для любого бинарного дерева D с k висячими вершинами*

$$T(D) \leqslant 10 \cdot k^{\log_2 \frac{4}{3}}.$$

Обозначим через $T_i'(D)$ сумму сложностей ярусов ребер с номерами от i до n в дереве D.

Лемма 3. *При $n \to \infty$, $k \to \infty$, $i \geqslant \log_2 k + \log_2 \log_2 (kn)$ для любого бинарного дерева D высоты n с k висячими вершинами выполнено*

$$T_i'(D) = \bar{o}\left(\left(\frac{k}{\log_2(kn)}\right)^{\log_2 \frac{4}{3}}\right).$$

Обозначим через $\chi^p_{k,l,k'}$ долю разбиений числа k на l слагаемых, p из которых в точности равны 1, а остальные $l-p$ не меньше k'.

Лемма 4. *При $n \to \infty$, $k \to \infty$, $2\log_2 \log_2 k \leqslant k' \leqslant \log_2 k$, $l \leqslant \frac{k}{\log_2 k}$,*

при $l \leqslant \sqrt{\frac{k}{(\log_2 k)^\alpha}}$, $\alpha > 1$ и $p > 0$,
при $\sqrt{\frac{k}{(\log_2 k)^\alpha}} \leqslant l \leqslant c\sqrt{k}$, $\alpha > 1$, $c \in \mathbf{R}$ и $p > \log_2 k$,
при $k = \bar{o}(l^2)$ и $p > 2\frac{l^2}{k}$ выполнено

$$\chi^p_{k,l,k'} = \bar{o}\left(\frac{1}{\log_2 k}\right).$$

Лемма 5.

$$\max_{k_1+\cdots+k_p=k} \prod \left(1 - \frac{1}{2^{k_i}}\right) = \left(1 - \frac{1}{2^{\frac{k}{p}}}\right)^p.$$

Лемма 6. *Пусть в усеченном бинарном дереве D, решающем задачу $I \in \mathcal{I}(n,k)$, на каждом ярусе вершин с номером большим i, два ребра выходят менее чем из $\frac{1}{3}$ вершин. Тогда $T(D) < 10 \cdot T_i(D)$.*

Доказательство лемм 2–6 приведено в работе [3].

Будем говорить, что подмножество V булева куба B_2^n принадлежит подкубу размерности 2^{n-1}, заданному координатой x_i, если для всех элементов множества V значения координаты с номером i совпадают. Обозначим через $\mu(n,k,i)$ долю k-элементных подмножеств булева куба размерности n, которые целиком содержатся в одном из двух подкубов, задаваемом координатой x_i.

Лемма 7. *Пусть $n = \bar{o}(2^k)$, $n \geqslant 4$. Тогда при $k > 2^{n-1}$ $\mu(n,k,i) = 0$; при $k < 2^{n-1}$*

$$\frac{2}{5^k} < \mu(n,k,i) \leqslant \frac{2}{2^k}.$$

Доказательство. Если размерность библиотеки больше, чем 2^{n-1}, то ни в каком подкубе она целиком находиться не может. Значит, при $k > 2^{n-1}$ $\mu(n,k,i) = 0$ для любого i. Для того, чтобы некоторая библиотека V принадлежала подкубу, задаваемому координатой x_i, необходимо, чтобы все ее элементы совпадали в координате x_i. Для фиксированного значения координаты можно выбрать $C^k_{2^{n-1}}$ подмножеств размера k в подкубе размерности $n-1$. Следовательно, доля библиотек, содержащихся в подкубе, заданном координатой x_i, равна $2\frac{C^k_{2^{n-1}}}{C^k_{2^n}}$.

$$\mu(n,k,i) = 2\frac{C^k_{2^{n-1}}}{C^k_{2^n}} = 2\frac{(2^{n-1}-k+1)\cdots 2^{n-1}}{(2^n-k+1)\cdots 2^n}.$$

Следовательно,

$$\mu(n,k,i) \leqslant 2\left(\frac{2^{n-1}}{2^n}\right)^k \leqslant \frac{2}{2^k}$$

и, поскольку

$$\frac{2^{n-2}-1}{2^n-1} = 1 - \frac{2^n - 2^{n-2}}{2^n-1} = 1 - \frac{3\cdot 2^{n-2}}{2^n-1} = 1 - \frac{3}{4-2^{2-n}} \geqslant \frac{1}{5}$$

при $n \geqslant 4$, то

$$\mu(n,k,i) \geqslant 2\left(\frac{2^{n-1}-k-1}{2^n-k-1}\right)^k \leqslant \frac{2}{5^k}.$$

Следовательно,

$$\frac{2}{5^k} < \mu(n,k,i) \leqslant \frac{2}{2^k}.$$

Лемма 7 доказана.

Обозначим через E^V_i число ребер на ярусе с номером i в дереве $A(V,\sigma)$, $i = 1,\ldots,n$. Тогда $T(A(V,\sigma)) = \sum_{i=1}^n \left(\frac{2}{3}\right)^{i-1} E^V_i$. Положим

$$E_i(k,n) = M_V E^V_i = \frac{\sum\limits_{V\in\mathcal{V}(n,k)} E^V_i}{C^k_{2^n}}.$$

Лемма 8. *При* $i \leqslant [\log_2 k - \log_2\log_2 k]$ *верно* $E_i(k,n) \sim 2^i$.

Доказательство. Пусть $\mathcal{D}_i(n,k)$ — все деревья, построенные алгоритмом A на библиотеках из множества $\mathcal{V}(n,k)$, содержащие на ярусе с номером i асимптотически 2^i ребер. Обозначим $D_i(n,k) = \frac{|\mathcal{D}_i(n,k)|}{C_{2^n}^k}$.

Для того, чтобы в каждом решающем дереве, построенном алгоритмом A по библиотеке $V \in \mathcal{V}(n,k)$, имела 2 ребра на первом ярусе, необходимо, чтобы она не содержалась ни в одном подкубе, заданном координатой x_1. В силу справедливости утверждения леммы 7

$$|\mathcal{D}_1(n,k)| \geqslant C_{2^n}^k - C_{2^n}^k \cdot \mu(n,k,1) > \left(1 - \frac{2}{2^k}\right) C_{2^n}^k \sim C_{2^n}^k.$$

Следовательно,

$$D_1(n,k) > 1 - \frac{2}{2^k} \sim 1 \text{ при } k \to \infty.$$

Значит,

$$E_1(n,k) = \frac{\sum\limits_{V \in \mathcal{V}(n,k)} E_1^V}{C_{2^n}^k} \geqslant \left(1 - \frac{2}{2^k}\right) \cdot 2 \sim 2 \text{ при } k \to \infty.$$

Предположим, что выполнено

$$D_{i-1}(n,k) > \left(1 - \bar{o}\left(\frac{1}{\log_2 k}\right)\right)^{i-1} \left(1 - \frac{2}{2^{k'}}\right)^{2^{i-1}}.$$

Разобьем все деревья из $\mathcal{D}_{i-1}(n,k)$ на классы эквивалентности следующим образом. Два дерева попадут в один класс, если их корневые поддеревья высоты $i-1$ полностью совпадают, в том числе совпадают нагрузки и нумерация ребер. Таким образом, в каждом классе находятся деревья с одинаковым началом. Берем произвольный класс. Пусть $l \sim 2^{i-1}$ — количество концевых ребер. Из деревьев класса возьмем только те, у которых число вершин, в которые проходит не менее $k' = \frac{\log_2 k}{\log_2 \log_2 \log_2 k}$ записей асимптотически равно $l \sim 2^{i-1}$ при $i \leqslant \log_2 k - \log_2 \log_2 k$. По утверждению леммы 4, при $i \leqslant \log_2 k - \log_2 \log_2 k$ доля таких деревьев составляет не менее чем $1 - \bar{o}\left(\frac{1}{\log_2 k}\right)$, так как

$$k' \cdot 2^i < \frac{\log_2 k}{\log_2 \log_2 \log_2 k} \cdot \frac{k}{\log_2 k} = \frac{k}{\log_2 \log_2 \log_2 k} = \bar{o}(k).$$

Для каждой вершины, в которую проходит более чем $\frac{\log_2 k}{\log_2 \log_2 \log_2 k}$ записей, вероятность того, что в вершине не реализуется подкуб размерности меньшей, чем $n-i$, заданный координатой x_i, не менее чем

$$1-\mu_2\left(n-i,\frac{\log_2 k}{\log_2\log_2\log_2 k},x_i\right)\geqslant 1-\frac{2}{\frac{k}{\log_2\log_2 k}}=1-\frac{2\log_2\log_2 k}{k}.$$

Следовательно, в каждом классе, доля деревьев из $\mathcal{D}_i(n,k)$ не меньше чем

$$\left(1-\bar{o}\left(\frac{1}{\log_2 k}\right)\right)\left(1-\frac{2\log_2\log_2 k}{k}\right)^{2^{i-1}}.$$

Значит и для всего множества $\mathcal{D}_i(n,k)$ выполнено

$$\begin{aligned}D_i(n,k)&>D_{i-1}(n,k)\left(1-\bar{o}\left(\frac{1}{\log_2 k}\right)\right)\left(1-\frac{2\log_2\log_2 k}{k}\right)^{2^{i-1}}>\\&>\left(1-\bar{o}\left(\frac{1}{\log_2 k}\right)\right)^{i}\left(1-\frac{2\log_2\log_2 k}{k}\right)^{2^{i}}>\\&>\left(1-\bar{o}\left(\frac{1}{\log_2 k}\right)\right)^{i}\left(1-\frac{2\log_2\log_2 k}{k}\right)^{\frac{k}{\log_2(kn)}}\sim 1\end{aligned}$$

при $i\leqslant \log_2 k-\log_2\log_2(kn)$. Значит, доля деревьев, построенных при помощи A на библиотеках из $\mathcal{V}(n,k)$, для которых усеченное бинарное решающее дерево имеет асимптотически 2^i ребер на ярусе с номером $i\leqslant \log_2 k-\log_2\log_2 k$, не меньше чем $D_i(n,k)\sim 1$ при $k\to\infty$. Тогда

$$\begin{aligned}E_i&=\frac{\sum\limits_{V\in\mathcal{V}(n,k)}\sum\limits_{\sigma}E_i^{(V,\sigma)}}{C_{2^n}^k}\geqslant\\&\geqslant\left(1-\bar{o}\left(\frac{1}{\log_2 k}\right)\right)^{i}\left(1-\frac{2\log_2\log_2 k}{k}\right)^{\frac{k}{\log_2(kn)}}2^i\sim 2^i\end{aligned}$$

при $k\to\infty$.

Лемма 8 доказана.

Обозначим через $\mathcal{D}'(n,k)$ долю таких деревьев, построенных при помощи A на библиотеках из $\mathcal{V}(n,k)$, что $T(D)\leqslant 66\frac{k}{\log_2 k}$.

Лемма 9. *При* $n \to \infty$, $k \to \infty$, $k = \bar{o}(2^n)$, $n = \bar{o}(2^k)$,

$$\mathcal{D}'(n,k) \succeq 1 - \frac{1}{\exp(k^{\varepsilon})}$$

для некоторого $0 < \varepsilon < \frac{\log_2 5}{3}$.

Доказательство. Положим $i_{\max} =]\log_2 k - \log_2\log_2 k + \log_2 9[$. Возьмем все деревья, построенные алгоритмом A на библиотеках из $\mathcal{V}(n,k)$, у которых на ярусе с номером $i_{\max}$ асимптотически $2^{i_{\max}} \geqslant \frac{9k}{\log_2 k}$ ребер. Все деревья разобьем на классы эквивалентности. Два дерева попадут в один класс, если корневые поддеревья высоты $i_{\max}$ одинаковы, в том числе одинаковы нагрузки и нумерация ребер, и если количество записей, проходящих в одинаковые вершины, является равным. Для каждого класса эквивалентности рассмотрим $p \geqslant \frac{3k}{\log_2(nk)}$ вершин на ярусе $i_{\max}$, в которые проходит максимальное количество — $k_{j_1}, \ldots, k_{j_p}$ — записей. Пусть в некоторую вершину v, из которой исходит дерево высоты $n' = n - i_{\max}$, проходит k' записей. Если все записи содержатся хотя бы в одном подкубе, размерности $n' - 1$, то у любого дерева, выходящего из вершины v и построенного для данных записей алгоритмом A, из корня будет выходить точно одно ребро. Значит, доля деревьев, имеющих точно одно исходящее ребро из вершины v, равна доле библиотек, принадлежащих некоторому подкубу, размерности, меньшей чем n'. Заметим, что число вершин на ярусе с номером $i_{\max}$, в которые проходит более $2^{n-i_{\max}-1}$ записей равна $\bar{o}(p)$, так как если бы такое число вершин равнялось бы $c \cdot 2^{i_{\max}}$, (c — константа), то

$$c \cdot 2^{i_{\max}} \cdot 2^{n-i_{\max}-1} = \frac{c}{2} \cdot 2^n > k > 2^{i_{\max}}$$

при $k = \bar{o}(2^n)$, что противоречит бинарности дерева. Тогда доля деревьев из класса, у которых из всех данных вершин выйдет два ребра, по лемме 7 и лемме 5 не больше чем

$$\prod_{i=1}^{p}\left(1 - \frac{2}{5^{k_i}}\right) < \left(1 - \frac{2}{5^{\frac{k}{p}}}\right)^{p-\bar{o}(p)} \sim \frac{1}{e^{5^{\frac{k}{p}}}}.$$

В свою очередь,

$$\frac{k}{p} < \frac{\log_2 k}{3},$$
$$5^{\frac{k}{p}} < k^{\frac{\log_2 5}{3}},$$
$$\frac{p}{5^{\frac{k}{p}}} > \frac{k^{1-\frac{\log_2 5}{3}}}{\log_2 k} > k^{\varepsilon} \to \infty$$

для некоторого $0 < \varepsilon < 1 - \frac{\log_2 5}{3} \approx 0{,}226$ при $k \to \infty$. Следовательно,

$$\frac{1}{\exp(\frac{p}{5^{k/p}})} \to 0$$

при $k \to \infty$. Значит, у более чем $1 - \frac{1}{\exp(k^{\varepsilon})}$ деревьев из класса доля вершин, из которых выйдет два ребра, не превосходит $\frac{3k}{\log_2 k}$, то есть трети числа всех ребер. Очевидно, что если D — произвольное дерево из $\mathcal{A}1(n,k)$, имеющее асимптотически $2^{i_{\max}}$ на ярусе с номером $i_{\max}$, то для любого яруса с номером $i \geqslant i_{\max}$ количество ребер не менее $\frac{9k}{\log_2 k}$, и с вероятностью более чем $1 - \frac{1}{\exp(k^{\varepsilon})}$, менее трети всех вершин яруса i имеют полустепень исхода, равную 2. Поскольку, в соответствии с утверждением леммы 3, для любого дерева сложность последних $n - \log_2 k - \log_2 \log_2 k$ ярусов ребер равна $\bar{o}\left(\left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}\right)$, то доля деревьев, удовлетворяющих условиям леммы 6, не менее

$$\left(1 - \frac{1}{\exp(k^{\varepsilon})}\right)^{2\log_2 \log_2 k} \sim 1 - \frac{2\log_2 \log_2 k}{\exp(k^{\varepsilon})}.$$

По лемме 6 сложность дерева D по порядку равна сложности первых $]\log_2 k - \log_2 \log_2 k + \log_2 9[$ ярусов. То есть

$$T(I) \leqslant 10 \cdot 2 \cdot \sum_{i=1}^{]\log_2 k - \log_2 \log_2 k + \log_2 9[} \left(\frac{4}{3}\right)^{i-1} \leqslant$$
$$\leqslant 20 \cdot \left(\frac{4}{3}\right)^{\log_2 9 + 1} \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}} < 66 \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}.$$

Заметим, что оценка числа деревьев с нужной нам сложностью не зависит от выбора класса. Следовательно, доля всех деревьев со сложностью не более $66\left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}$ при $n \to \infty$, $k \to \infty$, составляет не менее $1 - \frac{2\log_2\log_2 k}{\exp(k^\varepsilon)}$, $\varepsilon > 0$.

Лемма 9 доказана.

2.2. Доказательство теоремы

Поскольку $|\{V \in \mathcal{V}(n,k)\}| = C_{2^n}^k$, то

$$\bar{T}1(n,k) = \frac{\sum\limits_{V\in\mathcal{V}(n,k)} \bar{T}1(V)}{C_{2^n}^k} = \frac{\sum\limits_{V\in\mathcal{V}(n,k)} T(A(V,\sigma))}{C_{2^n}^k},$$

где σ — тождественная перестановка. Положим $i_{\max} = [\log_2 k - \log_2\log_2 k]$. Из утверждения леммы 8 следует, что $E_i(n,k) \sim 2^i$ при $i \leqslant i_{\max}$. Тогда при $n \to \infty$, $k \to \infty$

$$\bar{T}1(n,k) = \frac{\sum\limits_{V\in\mathcal{V}(n,k)} \sum\limits_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i^V}{C_{2^n}^k} = \frac{\sum\limits_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} \cdot \left(\sum\limits_{V\in\mathcal{V}(n,k)} E_i^V\right)}{C_{2^n}^k} =$$

$$= \sum_{i=1}^{n} \left(\frac{2}{3}\right)^{i-1} E_i \geqslant \sum_{i=1}^{i_{\max}} \left(\frac{2}{3}\right)^{i-1} E_i \sim \sum_{i=1}^{i_{\max}} \left(\frac{2}{3}\right)^{i-1} 2^i =$$

$$= \frac{3}{2} \sum_{i=1}^{i_{\max}} \left(\frac{4}{3}\right)^{i-1} \geqslant \frac{3}{2} \cdot 3 \left(\frac{4}{3}\right)^{\frac{k}{\log_2 k}-1} = \frac{27}{8} \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}.$$

С другой стороны, из утверждений леммы 9 и леммы 2 следует, что при $n \to \infty$, $k \to \infty$

$$\bar{T}1(n,k) \leqslant \left(1 - \frac{2\log_2\log_2 k}{\exp\left(k^\varepsilon\right)}\right) \cdot 66 \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}} +$$

$$+ \frac{10 \cdot k^{\log_2 \frac{4}{3}} \cdot 2 \cdot \log_2\log_2 k}{\exp(k^\varepsilon)} \sim 66 \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}.$$

Значит,

$$\bar{T}1(n,k) \asymp \left(\frac{k}{\log_2 k}\right)^{\log_2 \frac{4}{3}}.$$

Теорема доказана.

В заключение автор выражает благодарность Э. Э. Гасанову за постановку задачи и научное руководство.

Список литературы

[1] Блайвас Т. Д. Оптимальное решение задачи интервального поиска на булевом кубе в классе сбалансированных древовидных схем // Интеллектуальные системы. Т. 7, вып. 1–4.

[2] Блайвас Т. Д. Асимптотики задачи интервального поиска на булевом кубе в классе сбалансированных древовидных схем // Дискретная математика. Т. 16, вып. 4. 2004. С. 65–78.

[3] Блайвас Т. Д. Один алгоритм решения задачи интервального поиска на булевом кубе // Интеллектуальные системы. Т. 8, вып. 1–4.

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