DOI 10.37882/2223-2966.2024.9.22

ОБ ОДНОМ ПОДХОДЕ К ПОСТРОЕНИЮ ПОЛНЫХ КОДОВЫХ ДЕРЕВЬЕВ

ABOUT ONE APPROACH TO CONSTRUCTION OF COMPLETE CODE TREES

V. Mironkin
V. Sheikin
G. Astvatsatryan
G. Kochkonyan
M. Surkov

Summary. The article discusses one of the approaches to constructing complete code trees and the related problem of constructing optimal prefix codes. The assumption is formulated and proven that the value of the ratio of the number of terminal vertices to the total number of vertices in a complete D-ary tree tends to the value given by the formula: $P_{leaf} = 1 - \frac{1}{D}$, where P_{leaf} is the proportion of terminal vertices. A new algorithm for generating a prefix code has been developed, using the method of randomly selecting end vertices in a complete D-ary tree of arbitrary height n, as well as a test bench that visually implements this algorithm.

Keywords: code trees, complete code trees, prefix codes, algorithm for constructing prefix codes.

Миронкин Владимир Олегович
кандидат физико-математических наук, доцент, Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
vmironkin@hse.ru
Шейкин Всеволод Владимирович
Преподаватель, Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
vsheykin@hse.ru
Аствацатрян Георгий Леонович
Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
glastvatsatryan@edu.hse.ru
Кочконян Гарик Гарикович
Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
ggkochkonyan@miem.hse.ru
Сурков Максим Андреевич
Московский институт электроники и математики Национального исследовательского университета «Высшая школа экономики»
masurkov_1@edu.hse.ru

Аннотация. В статье рассматривается один из подходов к построению полных кодовых деревьев и связанная с этим задача построения оптимальных префиксных кодов. Формулируется и доказывается предположение о том, что значение отношения числа концевых вершин к общему числу вершин в полном D-арном дереве стремится к величине, заданной формулой: $P_{leaf} = 1 - \frac{1}{D}$, где P_{leaf} – доля концевых вершин. Разработан новый алгоритм для формирования префиксного кода, использующий метод случайного выбора концевых вершин в полном D-арном дереве произвольной высоты *n*, а также тестовый стенд, наглядно реализующий данный алгоритм.

Ключевые слова: кодовые деревья, полные кодовые деревья, префиксные коды, алгоритм построения префиксных кодов.

Теория кодирования является фундаментальной областью, играющей важную роль в различных аспектах информационных технологий. Одна из ключевых задач теории кодирования — построение оптимальных префиксных кодов. С помощью префиксных кодов можно эффективно реализовывать алгоритмы сжатия данных, хранить и искать строки или ключи в базах данных [1, 2], а также поиска IP-адресов, маршрутизации и фильтрации трафика [3]. В рамках настоящего исследования изучаются характеристики полных кодовых деревьев и методы построения префиксных кодов на их основе.

Из курса теории кодирования хорошо известно, что кодовые деревья используются для проверки свойств префиксности [4]. Идеей настоящей работы является обратный процесс — построение префиксного кода на основе имеющегося полного кодового дерева.

В первую очередь, было необходимо доказать теорему о предельном значении отношения P_{leaf} числа концевых вершин к общему числу вершин в полном D-арном дереве. Затем требовалось привести математическое обоснование эффективности вероятностного подхода к построению префиксных кодов по сравнению с известными детерминированными методами. Это включало в себя детальный анализ структуры полных кодовых деревьев и выявление особенностей их использования для формирования префиксных кодов с целью оптимизации процесса их построения. Также важной задачей было создание тестового стенда, предназначенного для наглядного изучения свойств формирования префиксных кодов.

В рамках проведенной работы разработан новый алгоритм, ориентированный на увеличение скорости генерации префиксных кодов. Особенность этого алгоритма заключается в исключении необходимости постоянной проверки свойств префиксности, что может значительно повысить его эффективность.

Рассмотрим метод построения префиксного кода путем случайного выбора вершин (и концевых, и промежуточных) из указанного дерева. В таком дереве каждая концевая вершина является словом префиксного кода, следовательно, совокупность его концевых вершин представляет собой префиксный код. При этом независимо от параметров кодового дерева, таких как глубина и количество потомков, отношение числа концевых вершин к общему числу вершин полного кодового дерева стремится сверху к $1 - \frac{1}{D}$, где D — количество потомков дерева.

Теорема

Пусть n — высота полного дерева, D — количество потомков, тогда вероятность выбора концевой вершины стремится к значению $1 - \frac{1}{D}$, где D — количество потомков.

Доказательство

Рассмотрим два пограничных случая, когда полное дерево минимально (максимально), то есть имеет наименьшее (наибольшее) возможное число вершин, и при этом выполняются условия полноты дерева.

В случае минимального дерева число концевых вершин равно $n(D-1)+1$, так как в дереве высоты n на каждом уровне, кроме последнего, находится ровно $D-1$ концевых вершин, а на последнем уровне — D концевых вершин. Получаем $n(D-1)+1$ — число концевых вершин минимального полного дерева. При этом общее число вершин равно $nD+1$, так как на каждом уровне всего D вершин и один корень дерева.

Рассмотрим отношение числа концевых вершин к общему числу вершин этого дерева. В этом случае доля концевых вершин равна $\frac{\left(n(D-1)+1\right)}{(nD+1)}$. В случае предельного соотношения

$$\lim_{n\to\infty}\frac{\left(n(D-1)+1\right)}{(nD+1)} = \lim_{n\to\infty}\frac{nD-n+1}{nD+1} = .$$

$$= 1 - \lim_{n\to\infty}\frac{n}{nD+1} = 1 - \frac{1}{D}$$

В случае максимального дерева число концевых вершин равно D^n, так как все концевые вершины расположены на последнем уровне дерева, и с каждым уровнем дерева количества концевых вершин увеличивается в D раз. На последнем уровне их количество равняется D^n.

Максимальное число концевых вершин равно D^n. Чтобы найти общее число вершин, необходимо воспользоваться формулой геометрической прогрессии, где $q=D$, а кол-во элементов прогрессии равняется $n+1$. Таким образом, общее число вершин составляет $\frac{D^{n+1}-1}{D-1}$. В случае предельного соотношения

$$\lim_{n\to\infty}\frac{D^n(D-1)}{D^{n+1}-1} = 1 - \lim_{n\to\infty}\frac{D^n-1}{D^{n+1}-1} = 1 - \frac{1}{D}.$$

Как можно заметить, в обоих случаях полученные значения совпадают, что и требовалось доказать.

Таким образом для того, чтобы реализовать алгоритм построения префиксного кода из N кодовых слов, достаточно реализовать построение полного кодового дерева, где общее число вершин K, умноженное на $1 - \frac{1}{D}$, будет превосходить N. Если данное условие выполняется, дерево гарантированно содержит N или более концевых вершин, так как $1 - \frac{1}{D}$ — оценка снизу. Суть алгоритма заключается в построении веток дерева до тех пор, пока общее число вершин K, умноженное на $1 - \frac{1}{D}$, не будет превосходить входной параметр необходимого числа кодовых слов N. Далее из концевых вершин формируется набор кодовых слов, который и является префиксным кодом.

Для эмпирической оценки сходимости предела была реализована функция для проверки значений с различными параметрами на языке функционального программирования Wolfram.

Полученное выражение можно интерпретировать как предельное значение вероятности P_{leaf} выбора концевой вершины при равновероятном распределении на множестве вершин D-арного дерева. При этом $P_{leaf} \to 1$ при $D \to \infty$.

Для реализации программного обеспечения было построено архитектурное решение, чтобы в дальнейшем иметь возможность портировать ядро программы на различные операционные системы [5]. В качестве ядра системы был выбран язык программирования Python, который позволяет портировать основной функционал приложения на платформы MacOS, iOS, Linux, Windows, Android. Для визуализации полученных результатов была выбрана библиотека Qt.

Результатом разработки тестового стенда стало Desktop-приложение. При разработке графического интерфейса первостепенной целью было создание удобного для пользователей инструмента, а также мультиплатформенность [6, 7, 8].

Возможности приложения охватывают весь цикл работы с префиксными кодами. Пользователь получает возможность определить параметры генерации дерева, вводя значения, такие как «количество концевых вершин», «глубина дерева», «количество потомков» (рис. 1), через интуитивно понятный интерфейс с соответствующими полями для ввода.

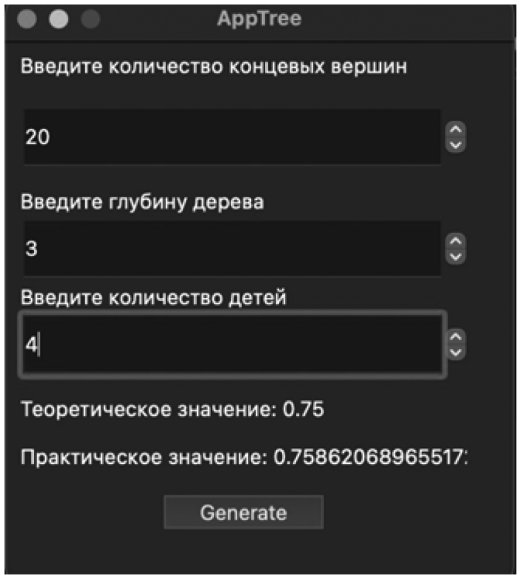

Рис. 1. Интерфейс приложения

Важным элементом функционала является предоставление доступа к dataset, содержащему готовые префиксные коды для каждой концевой вершины. Это позволяет не только визуально оценить результаты генерации, но и активно выбирать и анализировать подходящие префиксные коды для конкретных практических задач (рис. 2, 3).

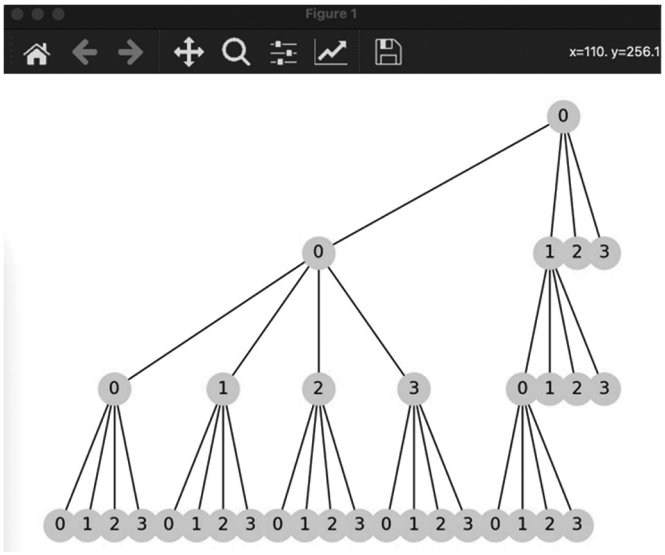

Рис. 2. Интерфейс приложения

Рис. 3. Интерфейс приложения

Для наглядной демонстрации эффективности разработанного метода была проведена оценка времени работы нашего алгоритма в условиях возрастающего количества концевых точек (график темно-серого цвета) в сравнении с аналогичным детерминированным (и, фактически, самым распространенным) алгоритмом генерации префиксных кодов — алгоритмом Хаффмана (график светло-серого цвета) (рис. 4). График показывает, что уже при генерации от 50000 концевых вершин разработанный алгоритм работает намного эффективнее.

Для ещё большей наглядности было проведено временное сравнение работы нашего и уже существующих

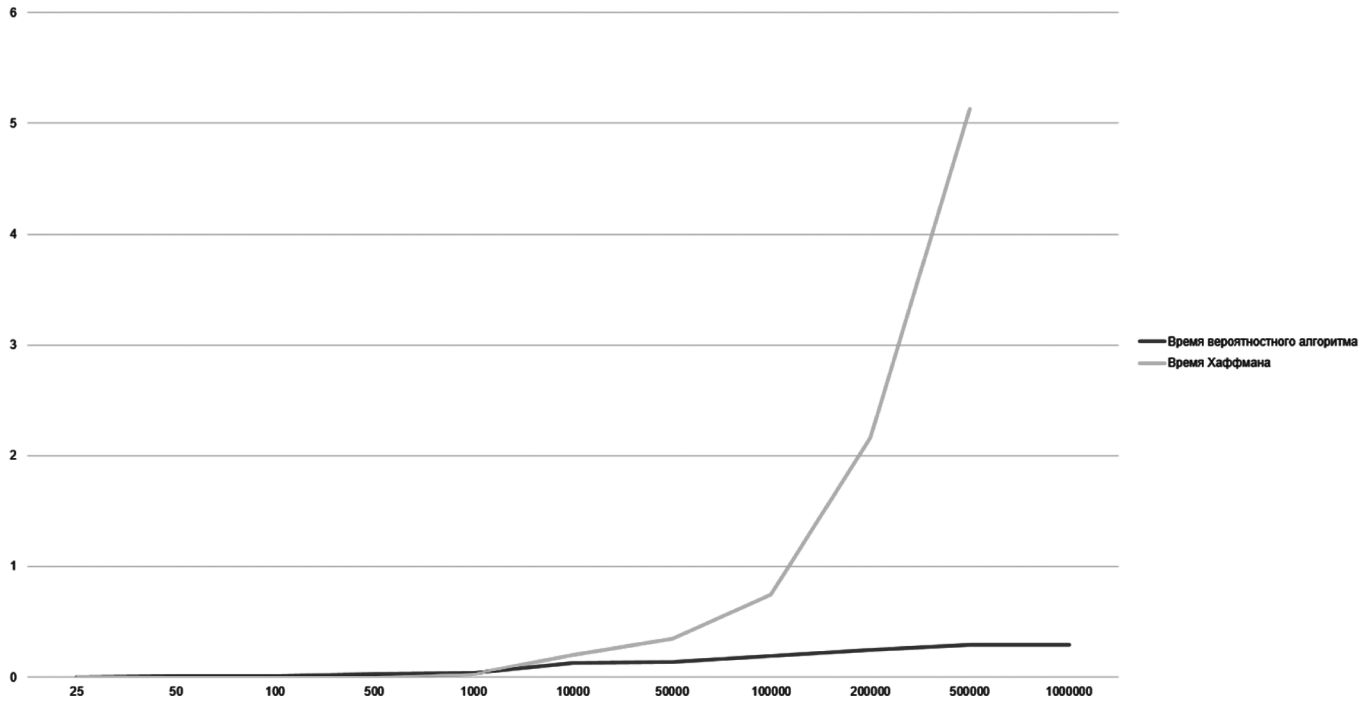

Рис. 4. Сравнение работоспособности детерминированных и разработанного алгоритма

Таблица 1.

Сравнение эффективности разработанного алгоритма в сравнении с детерминированными алгоритмами

Количество концевых вершин	Время генерации алгоритмом Хаффмана	Время генерации разработанным алгоритмом	Процентное соотношение
1000	0,12	0,13	–8 %
10000	0,3	0,21	42 %
50000	0,36	0,22	63 %
100000	0,8	0,24	333 %
200000	2,24	0,29	772 %
500000	5,15	0,31	1661 %

детерминированных алгоритмов (табл. 1). Очевидно, что при необходимости генерации больших кодов разработанный алгоритм именно ввиду новизны подхода к построению является гораздо более эффективным.

В статье разработан и теоретически обоснован новый вероятностный алгоритм построения префиксных кодовых деревьев. Доказана эффективность разработанного алгоритма с точки зрения времени в сравнении с классическими детерминированными методами генерации префиксных кодов. Разработано ПО, которое предоставляет пользователям эффективный и удобный инструмент для генерации кодовых деревьев, а также получения dataset с префиксными кодами. Проведенная адаптация указанного приложения обеспечивает широкие возможности в настройке и визуализации полученных деревьев, а также для анализа полученных данных.

ЛИТЕРАТУРА

1. МарьяновП.А.Уплотнениеструктурыданныхпрефиксногодереванаосновестатистическоймодели/П.А.Марьянов//Молодойученый.2016.№19(123).С.46–49.
2. Гудков А.С. Использование префиксных деревьев при построении систем анализа данных: дис. ... канд. физ.-мат. наук: 05.13.18 Москва, 2006 154 с. РГБ ОД, 61:07-1/472
3. Bernat V. IPv4 route lookup on Linux [Электронный ресурс] // Vincent Bernat blog: [2017]. URL: https://vincent.bernat.ch/en/blog/2017-ipv4-route-lookup-linux (дата обращения: 05.02.2024).
4. Теоретико-информационные аспекты защиты информации: [учебник] / Лось А.Б., Миронкин В.О. М.: URSS: Ленанд, 2023. 142 с.
5. Data Structures and Algorithms: Annotated Reference with Examples // R.L. Kruse, A.J. Ryba, Ch. L. Tondo. 2008. 26 с.
6. Rapid GUI Programming with Python and Qt» // M. Summerfield. 2007. 116 с.
7. Data Structures and Algorithms with Python» // K.D. Lee, S. Hubbard, 2015. 139 с.
8. Introduction to the Design and Analysis of Algorithms // A. Levitin. 2003. 366 с.

© Миронкин Владимир Олегович (vmironkin@hse.ru); Шейкин Всеволод Владимирович (vsheykin@hse.ru);
Аствацатрян Георгий Леонович (glastvatsatryan@edu.hse.ru); Кочконян Гарик Гарикович (ggkochkonyan@miem.hse.ru);
Сурков Максим Андреевич (masurkov_1@edu.hse.ru)
Журнал «Современная наука: актуальные проблемы теории и практики»