МЕТОДЫ ОБРАБОТКИ ИНФОРМАЦИИ
(Вычислительные системы)
1978 год Выпуск 74

УДК 8.74

ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ ЭФФЕКТИВНОСТИ ФУНКЦИЙ РАССТАНОВКИ

В.Д.Гусев, Т.Н.Титкова

I. Исходные понятия. Функция расстановки (аналог американскому термину "hashing function") в рамках данной статьи интерпретируется как элемент процедуры ассоциативного кодирования ("hash coding") [I] *). Под ассоциативным кодированием мы понимаем процедуру, осуществляющую отображение множества X ключей (имен, признаков), которыми снабжены элементы произвольного информационного массива, в некоторое подмножество A множества натуральных чисел, интерпретируемое как совокупность адресов, в соответствии с которыми элементы массива заносятся в оперативную или внешнюю память (каждому элементу - свой адрес).

Процедура должна быть достаточно простой в вычислительном отношении, экономичной по памяти (уточнение этих требований содержится в [I]) и универсальной в том смысле, что алгоритм вычисления адресов (алгоритм нумерации) не должен зависеть от того, какое конкретное множество ключей X, рассматриваемое как подмножество некоторого универсума U, предъявлено для нумерации.

Отображение X в A строится в два этапа. На первом этапе каждому $x \in X$ ставится в соответствие $h(x) \in A$, для чего, в принципе, может быть использована любая, удовлетворяющая сформулированным выше требованиям целочисленная функция (функция расстановки), определенная на U и обеспечивающая, по возможности, минимальное число наложений различных объектов. Под наложением понимается ситуация, когда двум или более различным ключам может быть поставлен в соответствие одинаковый адрес. Возникающие наложения алго -

*) В некоторых работах оба эти термина отождествляются.

ритмически устраняются на втором этапе либо путем поиска свободной позиции в расстановочном поле (используя, например, линейный просмотр или вычисляя для того же аргумента новую функцию расстановки), либо путем объединения налагающихся объектов в списки и выделения под них дополнительной памяти.

2. Анализ направлений экспериментальных исследований. Исследованию эффективности различных процедур ассоциативного кодирования посвящены работы [2,3]. Эффективность этих процедур определяется тремя факторами: а) выбором функции расстановки (желательно, чтобы уже на первом этапе число наложений было минимальным); б) выбором способа устранения наложений; в) структурой множества X (в принципе, почти для каждой функции расстановки можно подобрать "патологический" для нее массив ключей X из U, на котором эффективность процедуры будет очень низкой из-за большого числа наложений на первом этапе). Количественной характеристикой эффективности процедуры в целом обычно является среднее (по всем $x \in X$) число проверок (сравнений одного объекта с другим), требующихся для занесения объекта в память или для извлечения его оттуда.

В работах [2,3] основное внимание уделено исследованию способов устранения наложений при работе с большими информационными массивами, размещаемыми во внешней памяти (диски, барабаны). Зависимость результатов от структуры множества X нивелировалась путем усреднения исследуемых показателей по восьми различным массивам ключей. Ориентация на размещение и поиск объектов во внешней памяти обусловила тот факт, что вопросы выбора функции расстановки в указанных работах оказались недостаточно исследованными.

Действительно, основной целью этих работ являлся выбор такой схемы устранения наложений, которая минимизировала бы число обращений к внешним устройствам, поскольку время, затрачиваемое на одно обращение, превышает на один-два порядка как время вычисления функции расстановки (t_h), так и время просмотра одного звена списка (t_1). Минимизировать число обращений удается путем разбиения расстановочного поля на блоки, каждый из которых может содержать несколько элементов информационного массива. Элементы с одинаковым значением функции расстановки $h(x)$ последовательно располагаются при этом по мере их поступления в блоке с

номером h(x). Считывание информации с внешних устройств осуществляется блоками. Нетрудно видеть, что в предельном случае можно сделать размер блока близким к объему оперативной памяти, но время поиска нужного объекта в таком блоке может уже значительно превысить время обращения к внешнему устройству. Возникающая оптимизационная задача по выбору подходящего размера блока должна учитывать многие плохо формализуемые факторы, поэтому на практике она обычно решается эмпирически [3].

Рассмотренная выше схема устранения наложений не предъявляет высоких требований к выбору функции расстановки, поскольку использование "блочной" структуры при размещении и поиске объектов сознательно предполагает увеличение числа наложений (сверх тех, что обусловлены несовершенством самой функции расстановки). Экспериментальные данные, приведенные в [2,3], подтверждают, что с увеличением размера блока эффективность различных (по типу используемой функции расстановки) процедур ассоциативного кодирования становится примерно одинаковой, т.е. необходимости в детальном исследовании свойств функции расстановки не возникает.

3. Ц е л ь р а б о т ы. Настоящая работа в отличие от [2, 3] ориентирована на класс задач, в которых расстановочное поле размещается в о п е р а т и в н о й памяти. Разбиение расстановочного поля на блоки в данном случае уже не является эффективной процедурой устранения наложений, поскольку приводит лишь к неоправданному увеличению длины просматриваемого списка *). При блоках размера I выбор функции расстановки уже существенно влияет на эффективность процедуры ассоциативного кодирования в целом.

Цель работы заключается в экспериментальном исследовании э ф ф е к т и в н о с т и различных ф у н к ц и й р а с с т а н о в к и на массивах ключей из ограниченного, но широко используемого класса. Выявлены х а р а к т е р и с т и к и м а с с и в а к л ю ч е й X и параметры функций расстановки, влияющие на эффективность. Критерием для сравнения различных функций расстановки при фиксированном X являлся коэффициент заполнения

*) Исключение составляет ситуация, когда $t_h/t_1 \gg 1$. При этом оптимальной может оказаться структура расстановочного поля с блоками относительно небольшого размера (выгоднее просмотреть больший список, чем лишний раз вычислить функцию расстановки).

расстановочного поля

$$\beta(X,\alpha) = n_0/N\,|_{\alpha=n/N},$$

где α – коэффициент загрузки расстановочного поля, N – его объем, n_0 – число занятых позиций расстановочного поля, т.е. таких позиций, к которым произошло хотя бы по одному обращению в процессе заполнения расстановочного поля, n – число элементов в массиве X ($(n-n_0)$ – число наложений). Заметим, что параметры t_n и β не являются независимыми, поскольку добиться увеличения коэффициента заполнения β (при фиксированных X и α) удается, как правило, лишь за счет увеличения времени вычисления функции расстановки.

Схемы формирования адреса, реализуемые большинством известных функций расстановки, строятся по принципу максимального приближения к пуассоновской модели. В соответствии с этой моделью каждому ключу $x \in X$ (предполагается, что в массиве X все ключи различны) с одинаковой вероятностью может соответствовать любое значение адреса внутри расстановочного поля [I]. Если обозначить через μ_r ($r = 0,1,2,\ldots,n$) случайную величину, равную числу позиций расстановочного поля, к которым произошло ровно r обращений при заполнении поля, то для двух первых моментов этой величины при больших значениях n и N справедливы соотношения:

$$M\mu_r \sim N \cdot P_r,$$

$$D\mu_r \sim N P_r\left[1 - P_r - \frac{P_r}{\alpha}(\alpha - r)^2\right],$$

где

$$P_r = \frac{\alpha^r}{r!} e^{-\alpha}.$$

Учитывая, что $\beta = 1 - \mu_0/N$, для соответствующих моментов коэффициента заполнения получаем

$$\begin{aligned} M\beta_{\text{пуас}} &= 1 - M\mu_0/N \sim 1 - e^{-\alpha}, \\ D\beta_{\text{пуас}} &= \frac{1}{N^2} D\mu_0 \sim \frac{e^{-\alpha}}{N}(1 - e^{-\alpha} - \alpha e^{-\alpha}). \end{aligned} \qquad (I)$$

Значимое отклонение в меньшую сторону выборочного значения параметра β от оценки, вычисленной в соответствии с (I), характеризу-

ет меру несовершенства используемой функции расстановки на данном наборе ключей. Усредняя выборочные значения β по множеству наборов ключей из фиксированного класса и сравнивая β с $M_{\text{пас}}$, можно сделать выводы о целесообразности использования конкретной функции расстановки для ключей заданного класса.

4. Функции расстановки. В [2] представлен достаточно полный список используемых на практике функций расстановки (6 методов). Мы ограничим наше рассмотрение тремя из них (методы деления, свертки и середины квадрата), представляющими, с нашей точки зрения, наибольший интерес в плане достижения практически приемлемых значений параметров β и t_h.

Обозначим число разрядов в машинном слове через r и пронумеруем их слева направо (от старших разрядов к младшим), используя диапазон значений от 0 до r-1. Тогда код x ключа при вычислении на ЭВМ интерпретируется как целое число

$$K(x) = \sum_{i=0}^{r-1} x_i \cdot 2^{r-i-1}, \qquad (2)$$

где x_i – значения соответствующих двоичных разрядов этого кода.

Если длина кода x превышает размер машинного слова, осуществляется промежуточное отображение кода x в код x', длина которого соответствует машинному слову (например, путем использования операции группового циклического сложения). Все дальнейшие операции проводятся с кодом x' .

Если длина ключа меньше r , существуют различные варианты размещения ключа внутри машинного слова. В этом случае положение ключа будем фиксировать параметрами n_1 и n_2 , значения которых определяют номера первого и последнего разрядов кода.

4.1. Метод деления реализует вычисление остатка от целочисленного деления h(K) = K mod M , который интерпретируется как адрес элемента информационного массива, снабженного ключом x . Программное получение h(K) в проводившихся на ЭВМ "Минск-32" экспериментах осуществлялось по схеме

$$h(K) = K - M \cdot \text{entiere}\,(K/M), \qquad (3)$$

где для вычисления целой части от деления K на M использовалась стандартная программа из математического обеспечения ЭВМ. Варьируемым параметром для данного метода являлся размер расстановочного поля M.

4.2. Метод свертки в описываемых экспериментах реализовался следующим образом. Код x ключа сдвигался вправо на Δ_1 разрядов. Исходный и сдвинутый коды циклически складывались и из полученного кода со сдвигом на Δ_2 разрядов от правого конца выделялись m адресных разрядов [*]. Варьируемыми параметрами являлись величины сдвигов Δ_1 и Δ_2. В простейшем варианте метода свертки параметр Δ_2 полагался равным нулю.

4.3. Метод середины квадрата является модификацией предложенного фон Нейманом способа получения псевдослучайных равномерно распределенных чисел. Величина K(x) возводится в квадрат и из результата выделяются с р е д и н е m разрядов, определяющие адрес. Варьируемым параметром является величина сдвига Δ, характеризующая положение внутри машинного слова выделяемых разрядов (возрастанию Δ соответствует переход от младших разрядов к старшим).

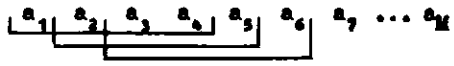
Поскольку при возведении в квадрат число разрядов исходного кода удваивается и может превысить размер машинного слова, для выделения адресных разрядов использовались либо только младшие разряды произведения (операция УЦ в системе команд ЭВМ "Минск-32"), либо только старшие (операция УФ). Использование комбинации тех и других потребовало бы фактически удвоения времени вычисления функции расстановки без существенных надежд на увеличение коэффициента β. Поэтому данный вариант не рассматривался.

4.4. Умножение на обратное число рассматривалось как альтернатива методу деления для машин, у которых операция умножения выполняется существенно быстрее чем операция деления. Процедура нахождения адреса включает в себя: а) перевод ключа K из целочисленного представления в представление в виде числа с плавающей запятой K_1 (K_1 может отличаться от K вследствие потери младших разрядов при резервировании места для порядка числа); б) получение целой части произведения $K_1 \cdot c$, где $c = 1/M$ вычисляется один раз перед заполнением расстановочного поля; в) вычисление $h(K) = K - entier(K_1 \cdot c)$.

5. Х а р а к т е р и с т и к и н а б о р о в к л ю ч е й. Исследовался класс ключей, представленный всевозможными l-грамами (связными подпоследовательностями из l символов) русского текста. Для примера в символьной последовательности

---

[*] В проводившихся с методами свертки и середины квадрата экспериментах использовались лишь значения $M = 2^m$.



выделены подчеркиванием три первых четырехграммы текста. Такой выбор множества ключей был продиктован как интересами самих авторов [4], так и широким распространением ключей данного типа в различных информационно-поисковых системах. Повторяющиеся 1-граммы исключались из массивов ключей специальными программными средствами. Для формирования различных наборов ключей использовались тексты трех видов - художественный, технический и общественно-политический, каждый длиной в $M = 10^5$ символов.

Символы алфавита представлялись семиразрядными двоичными кодами в стандартной кодировке ЭВМ "Минск-32". Одно машинное слово (37 разрядов) могло содержать не более 5 символов. Два младших разряда (35-й и 36-й) в силу специфики алфавитно-цифрового представления в данной ЭВМ не использовались. При $l > 5$ ключ занимал уже более одной ячейки памяти и использовался упоминавшийся выше механизм редукции длины ключа ($x \to x'$). При $l < 5$ незаполненными (если это не оговаривалось особо) оставались старшие разряды машинного слова.

Варьируемыми параметрами являлись: коэффициент загрузки рассановочного поля α (фактически он определяет число ключей в наборе), тип текста, длина ключа l, способ кодировки символов исходного алфавита, степень "рандомизации" ключей в наборе. Под рандомизацией здесь понимается некоторое предварительное преобразование ключа x (или x'), ориентированное на получение более равномерного (по сравнению с исходным) распределения значений ключей в заданном диапазоне.

Основное внимание было уделено исследованию трех последних параметров, не анализировавшихся в [2,3]. Забегая вперед, отметим, что различные эффекты, обусловленные вариацией этих параметров, оказывались весьма устойчивыми к изменению типа текста и коэффициента загрузки α. Это зачастую позволяло ограничиться для иллюстрации указанных эффектов фиксированным значением α и конкретным типом текста.

6. Экспериментальные результаты. Предварительное представление об исследуемых функциях расстановки дает табл. I, отражающая зависимость коэффициента заполнения β от коэффициента загрузки α при $l=5$ (общественно-политический текст).

Таблица I

Зависимость коэффициента заполнения расстановочного поля β от коэффициента загрузки α для различных методов (1 = 5)

α	Метод деления		Умножение на обратное число		Середина квадрата (УЦ)		Метод свертки		Пуассоновская модель	
	β	N	β	N	β	Δ	β	$\Delta_1(\Delta_2=0)$	Mβ	Dβ
0,5	0,4 0,37 0,40	8I92 8I9I 8I3I	0,05 0,30 0,34	8I92 8I9I 8I3I	0,I8 0,37 0,2I	II 23 25	0,24 0,I7 0,35	I3 2I 23	0,394	0,0026
0,7	0,05 0,47 0,5I	8I92 8I9I 8I3I	0,06 0,36 0,4I	8I92 8I9I 8I3I	0,2I 0,47 0,23	II 23 25	0,06 0,I9 0,44	3 2I 23	0,504	0,0030
0,9	0,05 0,55 0,60	8I92 8I9I 8I3I	0,06 0,4I 0,48	8I92 8I9I 8I3I	0,23 0,55 0,24	II 23 25	0,3I 0,2I 0,5I	I3 2I 23	0,594	0,0033
I,I	0,05 0,62 0,67	8I92 8I9I 8I3I	0,07 0,46 0,54	8I92 8I9I 8I3I	0,26 0,6I 0,24	II 23 25	0,07 0,22 0,56	3 2I 23	0,668	0,0035
t_h	II30 мксек		780 мксек		470 мксек		350 мксек			

В таблице представлены как минимальные (соответствующие неудачному выбору параметров функции расстановки), так и максимальные значения коэффициента заполнения. Для сравнения приведены соответствующие пуассоновские оценки. Нижняя строка таблицы дает ориентировочное представление о времени вычисления каждой функции расстановки (на ЭВМ "Минск-32"). Размер расстановочного поля варьировался для методов деления и умножения на обратное число, а для методов свертки и середины квадрата оставался фиксированным ($N = 8192$).

По результатам данной серии экспериментов можно отметить следующее.

1) Коэффициент заполнения β существенно зависит от выбора параметров N и Δ используемых функций расстановки. Максимальные значения коэффициента заполнения достаточно близки к значениям, получаемым из пуассоновской модели. Минимальные значения β могут быть во много раз меньше "пуассоновских" оценок. В частности, выбор делителя, кратного степени двойки (например, $N = 8192 = 2^{13}$), приводит к очень низким коэффициентам заполнения. Это объясняется тем, что при данном делителе значение $x \bmod N$ равно в соответствии с (3) содержимому 13 младших разрядов кода x. Поскольку в 13 разрядах не укладывается даже одна биграмма, а число р а з л и ч н ы х биграмм M_1 в тексте невелико (в общественно-политическом тексте, например, их оказалось всего 784), то при любом α коэффициент заполнения будет ограничен величиной M_1/N ($\approx 0{,}09$ - в данном случае).

К менее пагубным, но достаточно ощутимым последствиям может привести выбор в качестве делителя четного числа. В данном случае нежелательные эффекты возникают при нарушении баланса между числом четных и нечетных элементов в массиве ключей X и обусловлены они тем, что преобразование $x \bmod N$ при четном N сохраняет отношение четности (и нечетности) у элементов массива X. Из этого, например, следует, что если все $x \in X$ четны (нечетны), то при четном N коэффициент заполнения β при сколь угодно большом α не может быть выше 1/2. Заметим, что нарушение баланса "чет-нечет" может быть обусловлено не только структурой массива X, но и спецификой алфавитно-цифрового представления в используемой ЭВМ. К примеру, наличие двух нулевых младших разрядов в ЭВМ "Минск-32" приводит к тому, что все значения $K(x)$ в соответствии с (2) кратны 4. Сдвиг кода вправо на два разряда устраняет этот эффект. К примеру, при

M = 8184 коэффициент заполнения при α = 1,1 составляет 0,25 (без сдвига кода) и 0,64 (после сдвига кода). Данный пример показывает, что если длина ключа меньше размера машинного слова, размещение ключа внутри машинного слова не должно быть произвольным.

Хорошие результаты были получены при выборе в качестве делителей простых или составных (но с достаточно большими сомножителями) чисел (8131 = 173·47).

Некоторые аномалии наблюдаются и в этом случае (в частности, при M = 8191), но они будут пояснены ниже.

Заметим, что рекомендация о нежелательности использования составных делителей с малыми сомножителями не носит абсолютного характера. К примеру, для M = 4195, 4196, 4197, делящихся соответственно на 5,2,3, были получены при α = 1,1 коэффициенты заполнения 0,68, 0,67, 0,69, даже несколько превышающие соответствующую пуассоновскую оценку (коды предварительно сдвигались на 2 разряда). В то же время при использовании восьмиразрядной кодировки символов кратность делителя тройке привела бы в силу соотношения $10^n \bmod 3 = 4^n \bmod 3 = 1$ (n – положительное целое число) к наложению всех 1-грамм, отличающихся друг от друга лишь по порядку следования символов (при одинаковом их составе) [6].

2) Умножение на обратное число в общем случае не может рассматриваться как достаточно хорошая альтернатива методу деления. Хотя затраты на вычисление адреса и уменьшаются по сравнению с методом деления, но падает и коэффициент заполнения (вследствие частичной потери точности при переводе ключа из целочисленного представления в представление с плавающей запятой, а также при вычислении $c = 1/M$). Отметим попутно, что использование для получения $x \bmod M$ табличного алгоритма [5] позволяет сократить время вычислений не менее чем в полтора раза, т.е. сглаживаются различия между двумя методами и по параметру t_h .

3) Исследуемые методы ранжируются по параметрам t_h и β в обратном порядке. Наибольшие значения β достигаются, как правило, при использовании методов с максимальным значением параметра t_h , т.е. ни один из методов не является оптимальным одновременно по обоим параметрам (в смысле достижения максимального β при минимальном t_h).

6.1. Зависимость β от параметров сдвига иллюстрируется таблицами 2 и 3. Для значений l = 4,5,10 используется коэффициент загрузки α = 1,1. При l = 3 и M = 8192 добиться такого коэффициента

загрузки не удалось ввиду ограниченности числа различных триграмм в использовавшихся текстах. Фактические коэффициенты загрузки при указанных значениях параметров l и N составили: для общественно-политического текста $\alpha_{пол} = 0{,}65$, для художественного - $\alpha_{худ} = 0{,}83$, для технического - $\alpha_{техн.} = 0{,}62$. Соответствующие им пуассоновские оценки значений параметра β равны $M\beta_{пол.} = 0{,}48$, $M\beta_{худ.} = 0{,}56$ и $M\beta_{техн.} = 0{,}46$.

Таблица 2

Зависимость коэффициента заполнения β от параметров сдвига Δ_1 и Δ_2 для метода свертки (N = 8192)

Δ_1	Δ_2	$\beta_{худ.}$ (l=3)	$\beta_{худ.}$ (l=4)	$\beta_{худ.}$ (l=5)	$\beta_{пол.}$ (l=5)	Δ_1	Δ_2	$\beta_{пол.}$ (l=5)
I	0	0,08	0,07	0,07	0,07	I6	0	0,47
3	0	0,I0	0,08	0,07	0,07	I8	0	0,5I
5	0	0,28	0,20	0,I8	0,I7	20	0	0,42
7	0	0,I9	0,I6	0,I6	0,I4	I6	2	0,50
9	0	0,58	0,39	0,34	0,3I	I8	2	0,58
II	0	0,30	0,39	0,34	0,32	20	2	0,52
I3	0	0,34	0,4I	0,37	0,34	I6	4	0,52
I5	0	0,2I	0,37	0,33	0,3I	I8	4	0,56
I7	0	0,26	0,60	0,5I	0,48	20	4	0,45
I9	0	0,20	0,49	0,50	0,50	I6	6	0,52
2I	0	0,09	0,20	0,22	0,22	I8	6	0,52
23	0	0,06	0,28	0,58	0,56	20	6	0,42

По результатам данной серии экспериментов можно сделать следующие выводы.

I) При длине ключа, меньшей размеров машинного слова, метод свертки (в описанной выше модификации, т.е. с двукратным наложением) весьма неустойчив к изменению параметра сдвига Δ_1. Об этом свидетельствует наличие резких скачков коэффициента заполнения для близких значений Δ_1. В целом наблюдаемые значения β существенно уступают соответствующим пуассоновским оценкам, хотя возможны исключения (для l = 3 при $\Delta_1 = 9$ $\beta_{худ.} = 0{,}58 > M\beta_{пуас.} = 0{,}56$). Вариация положения внутри машинного слова выделяемых адресных разрядов (т.е. использование ненулевых значений параметра Δ_2) может привести к улучшению результатов, но объем перебора при этом намного возрастает.

Таблица 3

Зависимость коэффициента заполнения от параметра сдвига Δ для двух вариантов метода середины квадрата (УЦ и УФ; N = 8192)

Первый вариант метода середины квадрата (УЦ)					Второй вариант метода середины квадрата (УФ)				
l	Δ	$\beta_{пол.}$	$\beta_{худ.}$	$\beta_{техн.}$	l	Δ	$\beta_{пол.}$	$\beta_{худ.}$	$\beta_{техн.}$
3	3	0,01	0,04	0,04	4	0	0,46	0,47	0,46
	7	0,12	0,13	0,12		I	0,42	0,43	0,41
	II	0,41	0,47	0,38		2	0,39	0,40	0,38
	I5	0,48	0,56	0,45		3	0,34	0,35	0,34
	I7	0,48	0,56	0,46		4	0,29	0,30	0,29
	I9	0,48	0,57	0,46		5	0,24	0,24	0,24
	2I	0,48	0,56	0,46		II	0,04		
	23	0,48	0,57	0,46		2I	2^{-12}		
4	3	0,04	0,04	0,04	5	0	0,67	0,67	0,66
	7	0,10	0,10	0,10		I	0,67	0,67	0,66
	II	0,32	0,32	0,30		2	0,67	0,68	0,66
	I5	0,49	0,49	0,47		3	0,66	0,66	0,65
	I7	0,59	0,59	0,58		4	0,65	0,65	0,64
	I9	0,65	0,64	0,64		5	0,64	0,63	0,63
	2I	0,67	0,66	0,66		II	0,50		
	23	0,67	0,67	0,67		2I	0,13		
5	3	0,03	0,04	0,03	I0	5	0,66	0,66	0,66
	7	0,09	0,09	0,09		7	0,67	0,66	0,66
	II	0,26	0,28	0,25		9	0,65	0,66	0,65
	I5	0,40	0,43	0,40		II	0,66	0,65	0,64
	I7	0,48	0,51	0,47		I3	0,64	0,66	0,64
	I9	0,52	0,54	0,51		I5	0,63	0,65	0,64
	2I	0,56	0,59	0,56		I7	0,62	0,65	0,63
	23	0,61	0,64	0,62		2I	0,50	0,54	0,51

2) Несогласованность значений параметра сдвига Δ_1 (при $\Delta_2 = 0$) с длиной ключа (см. первый столбец табл. 2, $l = 3, \Delta_1 > 13$) и его расположением внутри машинного слова может привести к низким значениям β. К примеру, при $\Delta_1 = 21$ 35-й разряд, используемый в качестве адресного, всегда, т.е. при любых ключах с $l \le 5$, оказывается нулевым. Это объясняется тем, что указанный разряд является нулевым как в исходном коде (в силу специфики алфавитно-цифрового представления - см. п.5), так и в сдвинутом (стандартная кодировка символов такова, что первый разряд каждого символа является нулевым; при $\Delta_1 = 21$ первый разряд третьего символа соответствует 35-му разряду в сдвинутом коде). Аналогичные эффекты наблюдаются при $\Delta_1 = 14$ и 7.

3) В методе середины квадрата (как УЦ, так и УФ) зависимость β от Δ носит монотонный характер, т.е. при использовании операций типа УЦ коэффициент заполнения увеличивается с ростом Δ (по мере приближения к "середине квадрата"), а при использовании операции типа УФ - уменьшается (по мере удаления от "середины квадрата"). Максимальные значения коэффициента заполнения очень хорошо согласуются с пуассоновскими оценками (вариант с УЦ) при значении параметра сдвига $\Delta_c = r - n_1 - [m/2]$, которое обеспечивает точное совпадение положения выделяемых адресных разрядов с "серединой квадрата". Максимально допустимое для метода УЦ значение параметра сдвига, не приводящее к появлению нулевых разрядов в адресном коде, равно $r - m$. Отсюда видно, что лишь при значениях $n_1 > [m/2]$ в качестве адресных разрядов могут действительно быть выбраны средние разряды произведения. Данное условие выполняется для $l = 3,4$ (при $m \le 13$), но не выполняется для $l = 5$. Этим объясняется относительное ухудшение результатов в методе, использующем операцию УЦ, при $l = 5$.

4) При использовании операции УФ, фиксирующей только старшие разряды произведения, наблюдается резкое ухудшение результатов в случае стандартного расположения коротких ключей ($l = 3,4$) в младших разрядах машинного слова. Это объясняется тем, что средние разряды произведения в этом случае для нас недоступны (даже при $\Delta = 0$). При размещении коротких ключей в старших разрядах машинного слова результаты хорошо согласуются с пуассоновскими оценками при $\Delta_c = r - (n_2 - n_1 + 1) - [m/2]$. Ниже приведены соответствую-

ние данному случаю выборочные значения β как функции от l и Δ ($\alpha = 0{,}65$ для $l = 3$ и $\alpha = 1{,}1$ для $l = 4$; $m = 13$).

Таблица 4

Δ	β ($l=3$)	β ($l=4$)
0	0,17	0,67
1	0,24	0,67
2	0,31	0,67
3	0,38	0,66
4	0,43	0,66
5	0,45	0,66
6	0,47	0,67
7	0,48	0,67
8	0,49	0,66
9	0,49	0,66
10	0,48	0,64
11	0,48	0,62
12	0,48	0,57

При использовании длинных ключей ($l \geq 5$) в принципе невозможно точно выделить средние разряды произведения, используя только одну операцию УФ, однако выборочные значения β совпадают с пуассоновскими оценками (при $\Delta = 0$). Сопоставление вариантов ($l = 5$, УФ) и ($l=5$, УЦ) говорит о том, что зависимость β от Δ не совсем симметрична относительно "середины квадрата". Спад кривой в сторону старших разрядов происходит м е д л е н н е е, чем в сторону младших, т.е. использование операции УФ дает выигрыш в коэффициенте заполнения.

5) Сопоставление вариантов ($l = 5$, УФ) и ($l = 10$, УФ) показывает, что убывание β как функции от Δ во втором случае происходит гораздо медленнее, чем в первом. Поскольку второй вариант отличается от первого лишь более равномерным перемешиванием разрядов (за счет промежуточного отображения $x \to x'$), можно заключить, что предварительная рандомизация ключей повышает устойчивость метода.

6.2. Зависимость β от длины ключа для разных методов иллюстрируется табл.5. Параметры N (для метода деления) и Δ (для методов свертки и середины квадрата) подобраны оптимальным образом (за исключением делителя $N = 8191$, иллюстрирующего интересную аномалию).

Анализируя таблицу, нетрудно заметить, что длина ключа является одним из основных факторов, определяющих эффективность метода. При больших значениях l ($l = 10, 15$) все исследуемые методы на разных массивах ключей дают стабильные результаты, близкие к пуассоновским оценкам. Как и в предыдущем случае (см.п.6.1), это объясняется эффектом рандомизации ключей, возникающим при отображении $x \to x'$.

Действительно, в методе свертки вместо исходной модификации с двойным наложением получаем модификация с наложениями кратности

Таблица 5

Зависимость коэффициента заполнения β от длины ключа l (α = I,I за исключением l = 3, M = 8I9I,8292)

Метод		l	$\beta_{пол.}$	M(или Δ)	$\beta_{худ.}$	M(или Δ)	$\beta_{техн.}$	M(или Δ)
Деление		3	0,68 0,29	4079 8I9I	0,66 0,34	4079 8I9I	0,67 0,30	4079 8I9I
		4	0,67 0,55	4079 8I9I	0,66 0,56	4079 8I9I	0,67 0,58	4079 8I9I
		5	0,66 0,62	4079 8I9I	0,66 0,63	4079 8I9I	0,67 0,63	4079 8I9I
		I0	0,66	4079	0,67	4079	0,66	4079
		I5	0,67	4079	0,67	4079	0,67	4079
Середина квадрата (УЦ)	m =I3	3	0,48	I5-I9	0,57	I9,23	0,46	I7-23
		4	0,67	2I-23	0,67	23	0,67	22,23
		5	0,6I	23	0,64	23	0,62	23
	m =I2	I0	0,66	I9-23	0,67	I9	0,66	23
		I5	0,67	23,24	0,67	I7-2I	0,67	I7-23
Свертка	m=I3	3	0,49	9	0,58	9	0,47	9
		4	0,59	I6	0,60	I7	0,60	I6
		5	0,56	23	0,58	23	0,57	23
	m=I2	I0	0,65	I8	0,66	I7,I9	0,66	I7
		I5	0,67	I8,22	0,67	I3,I5,I9	0,66	I7-I9

три (l = I0) и четыре (l = I5). Улучшение результатов при переходе от l = 5 к l = I0,I5 в методе середины квадрата (УЦ) объясняется уменьшением за счет промежуточного отображения числа кодов, содержащих большое число нулевых разрядов (для такого рода ключей метод середины квадрата, как известно, довольно неэффективен). Наконец, устранение с ростом l аномальных эффектов в методе деления (см. M = 8I9I) объясняется как рандомизирующими свойствами отображения $x \to x^l$, так и увеличением отношения $l/]\log_2 M[$.Поясним данный эффект более подробно.

В экспериментах с методом деления было замечено, что при значениях $M = 2^m \pm k$, где k – малое целое положительное число, зачастую наблюдается значимое ухудшение результатов по сравнению с пуассоновской моделью даже в случаях, когда соответствующее M оказывается простым числом. Эффект выражен тем сильнее чем меньше от-

ношение $1/]\log_2 N[$. Для иллюстрации в табл.6 приведены выборочные значения β для различных $N = 2^m \pm K$ ($m = 7-13$). Диапазон простых делителей в интервале от 2^7 до 2^8 прослежен полностью. Коэффициент заполнения α равен I,I для всех значений l и N, кроме $l = 3$, $N = 8191 - 8193$, где $\alpha = 0{,}65$ (соответственно $M\beta_{пуас.} = 0{,}67$ в первом случае и 0,48 – во втором).

Наибольшие абсолютные отклонения от пуассоновских оценок (исключая рассмотренные выше случаи, когда $N = 2^m$) наблюдаются при $l = 2$, $m = 7-9$; $l = 3$, $m = 7, 13$. Для пояснения этих эффектов воспользуемся легко интерпретируемым соотношением

$$x \bmod(2^m \pm K) = (x \bmod 2^m \mp K \cdot [x : 2^m]) \bmod(2^m \pm K), \qquad (4)$$

которое можно рассматривать как преобразование, редуцирующее длину ключа x . Привязка к основанию 2^m в правой части (4) позволяет наглядно трактовать оба слагаемых в круглых скобках. Первое слагаемое – это m младших разрядов ключа x , второе слагаемое (при $k = I$) – оставшиеся $(71 - m)$ старших разрядов. При $m = 13$ и $l = 3$ мы не получаем даже двукратного наложения разрядов во в с е м "адресном" диапазоне (наложение затрагивает лишь 7 младших разрядов). Этим и объясняется резкое ухудшение результатов при $N = 8191$, 8193 с уменьшением l .

При $l = 2, 3$ и $N = 2^7 \pm 1$ последовательной редукцией длины ключа в соответствии с (4) достигаем двукратного наложения разрядов в адресном диапазоне для $l = 2$ и трехкратного для $l = 3$, чего в принципе было бы достаточно для получения удовлетворительных результатов. Однако совпадение значности кодировки (семь) со значением параметра m , характеризующим размер расстановочного поля, выявляет в данном случае некоторые дефекты стандартной и з б ы т о ч н о й кодировки, в частности, наличие кода 01 в первых двух разрядах каждого алфавитного символа. Целенаправленное изменение кодировки (см. п.6.3) значительно улучшает результат.

При $l = 2$, $m = 8-9$ вновь "работает" эффект малости отношения $1/]\log_2 N[$. При увеличении $l (l = 3-5)$ результаты резко улучшаются.

6.3. Зависимость коэффициента заполнения от кодировки символов исходного алфавита. Выше уже было показано, что избыточность кодировки, проявляющаяся в постоянстве значений определенных разрядов для в с е х кодов алфавита, может привести при неудачном выборе параметров функции расстановки к неудовлетворительным ре-

Таблица 6

Иллюстрация аномальных эффектов в методе деления

l = 2	N	I27	I29	255	256	257	5II	5I2	5I3
	β	0,46	0,47	0,43	0,23	0,43	0,37	0,26	0,37
l = 3	N	I27	I28	I29	I3I	I37	I39	I49	I5I
	β	0,5I	0,23	0,55	0,65	0,63	0,70	0,70	0,70
	N	I57	I63	I67	I73	I79	I8I	I9I	I93
	β	0,66	0,66	0,68	0,69	0,69	0,66	0,65	0,67
	N	I97	I99	2II	223	227	229	233	239
	β	0,7I	0,68	0,67	0,7I	0,69	0,65	0,70	0,66
	N	24I	25I	255	256	257	5II	5I2	5I3
	β	0,70	0,65	0,6I	0,20	0,57	0,69	0,20	0,65
	N	I023	I024	I025	2047	2048	2049	4093	4095
	β	0,64	0,I9	0,62	0,63	0,I8	0,62	0,62	0,59
	N	4096	4097	8I3I	8I9I	8I92	8I93		
	β	0,I6	0,58	0,48	0,29	0,I0	0,30		
l = 4	N	I27	I28	I29	255	256	257	5II	5I2
	β	0,57	0,23	0,53	0,64	0,20	0,66	0,66	0,I9
	N	5I3	I023	I024	I025	2047	2049	4093	4095
	β	0,67	0,64	0,I7	0,65	0,63	0,62	0,64	0,6I
	N	4096	4097	8I9I	8I92	8I93			
	β	0,I2	0,6I	0,55	0,08	0,54			
l = 5	N	I27	I28	I29	255	256	257	5II	5I2
	β	0,57	0,23	0,57	0,65	0,2I	0,69	0,68	0,I9
	N	5I3	I023	I024	I025	2047	2049	4093	4095
	β	0,70	0,68	0,I7	0,67	0,65	0,66	0,68	0,66
	N	4096	4097	8I9I	8I92	8I93			
	β	0,I2	0,67	0,62	0,07	0,63			

зультатам. Ниже будет рассмотрен еще один аспект, связанный с выбором кодировки. Стандартная кодировка символов алфавита (ГОСТ 10859-64), вообще говоря, лишь случайно может оказаться согласованной с частотными свойствами массива ключей, используемых для адресации. К примеру, если символам алфавита, наиболее часто

встречающимся в наборе ключей, присвоить коды с минимальным числом единиц (или нулей), можно сильно нарушить баланс нулей и единиц в целом по всему массиву ключей. Это, в свою очередь, может привести к увеличению числа наложений, т.е. к уменьшению β.

Влияние эффекта, обусловленного различными вариантами кодировки символов алфавита, проверялось на массивах ключей, составленных из 1-грамм политического текста. Использовались три типа кодировок (0,1,2).

Кодировке типа 0 соответствовала стандартная кодировка по ГОСТу. Баланс нулей и единиц в исходном тексте длиной $M = 10^5$ символов был нарушен при данной кодировке в сторону нулей ($f_0 - f_1 = \frac{1}{6}$, где f_0 и f_1 – относительные частоты встречаемости нулей и единиц в массиве).

Кодировка типа 1 выбиралась из условия сохранения баланса между числом нулей и единиц в исходном тексте ($f_0 \approx f_1$). Для этого символы текста упорядочивались по убыванию частоты встречаемости и последовательно кодировались таким образом, чтобы суммарный текущий баланс нулей и единиц у закодированных символов с учетом частоты встречаемости их в тексте оставался примерно одинаковым.

Кодировка типа 2 выбиралась из условия достижения максимального дисбаланса между числом нулей и единиц в тексте ($f_0 - f_1 \approx 0,6$). Для этого, как и в предыдущем случае, символы алфавита упорядочивались по убыванию частоты встречаемости. Затем первым семи символам в упорядоченном ряду присваивались коды 0000001, 0000010, ..., 1000000; следующим 21 символам (C_7^2) присваивались коды, содержащие по две единицы; наконец, оставшимся шести символам присваивались коды, содержащие по три единицы.

Результаты эксперимента приведены в табл.7 (общественно-политический текст). Для иллюстрации отобраны лучшие результаты по методам свертки и середины квадрата. Метод деления иллюстрируется на "аномальных" делителях.

По результатам эксперимента отметим следующее.

1) Наилучшими свойствами обладает кодировка типа 1, наихудшими – типа 2. В среднем при $l = 3,4,5$ вторая кодировка проигрывает первой по параметру β в полтора раза. По отношению к кодировке типа 0 кодировка типа 1 сглаживает аномальные эффекты в методах деления ($l = 3,4,5$; $N = 127,8191$) и свертки ($l = 10,15$; $\Delta_1 = 23$).

2) Использование перекодировки целесообразно в ситуациях, когда внеочередные значения β существенно уступают пуассоновским

оценкам. Если $\hat{\beta} \approx M\beta_{пуас}$, перекодировка не улучшит результата, а может лишь расширить диапазон значений параметров функции расстановки, для которых достигается оптимальный результат.

Таблица 7

Влияние кодировки символов алфавита на коэффициент заполнения ($\alpha = I,I$, кроме вариантов с $l = 3$, $N > 8000$)

Тип кодировки	l	Деление		Свертка (m=I3)		Середина квадрата (УЦ) (m = I3)	
		β	N	β_{max}	$\Delta_1(\Delta_2=0)$	β_{max}	Δ
0	3	0,5I 0,29	I27 8I9I	0,49	9	0,48	I5-I9
	4	0,55	8I9I	0,59	I6	0,67	2I-23
	5	0,62	8I9I	0,56	23	0,6I	23
	I0	0,66	4093	0,65	I8	0,66	I9-23
	I5	0,67	4093	0,67	I8	0,67	23,24
I	3	0,60 0,40	I27 8I9I	0,48	9	0,49	I7,23
	4	0,65	8I9I	0,64	I7	0,67	23
	5	0,66	8I9I	0,59	23	0,6I	23
	I0	0,67	4093	0,67	I7,I9,23	0,67	2I,24
	I5	0,67	4093	0,67	I9,2I,23	0,67	I7-24
2	3	0,24	8I9I	0,3I	9	0,4I	I7,2I
	4	0,45	8I9I	0,38	I5,I7	0,59	23
	5	0,53	8I9I	0,40	23	0,53	23
	I0	0,67	4093	0,63	23	0,68	23
	I5	0,67	4093	0,66	I9,23	0,66	I9,23

6.4. Зависимость β от степени "рандомизации" ключей. Рандомизация ключей (см. п.5) для каждого из трех исследуемых методов осуществлялась путем сдвига исходного кода x (или x') на фиксированное число разрядов и циклического сложения полученного кода с исходным. Очевидно, что процедуру рандомизации с последующим вычислением функции расстановки можно рассматривать как суперпозицию двух "элементарных" функций расстановки - свертки и исследуемой функции. Рандомизация применительно к самому методу свертки просто увеличивает на единицу кратность наложений по отношению к исходной модификации метода. Кратность наложений может служить ко-

личественной мерой степени рандомизации ключей. К примеру, в варианте метода свертки с предварительной рандомизацией ключей кратность наложений при $l = 15$ равна 5 (при соответствующим образом подобранных параметрах сдвига).

При рандомизации, предшествующей методам деления и середины квадрата, осуществлялся сдвиг кода вправо на 9 разрядов. Для метода свертки, в котором исходный код сдвигался вправо на Δ_1 разрядов, рандомизирующий сдвиг делался влево на Δ_1 разрядов. Параметр Δ_2 выбирался равным Δ_1. Диапазон значений Δ_1, обеспечивающий тройное наложение кодов по всем адресным разрядам, задается при этом неравенством $\Delta_1 \le \frac{1}{2}(n_2 - n_1 - m)$, т.е. с учетом того, что $n_1 = 7 \cdot (5-1)$, $n_2 = r - 3$, имеем для $m = 13$ и $l = 3,4,5$ соответственно $\Delta_{1max} = 4,7,11$.

Таблица 8

Зависимость β от степени рандомизации

Режим	l	Деление		Свертка (m = 13,12)		Середина квадрата УЦ (m = 13, 12)	
		β	N	β_{max}	Δ_1	β_{max}	Δ
Без рандомизации	3	0,51	127	0,49	9	0,48	15
		0,29	8191	0,29	13	0,48	19
	4	0,08	8192	0,59	16	0,67	21,23
		0,55	8191	0,48	12	0,66	19
	5	0,25	8184	0,56	23	0,61	23
		0,62	8191	0,51	18	0,58	22
	10	0,66	4093	0,65	18	0,66	19–23
	15	0,67	4093	0,67	18,22	0,68	23
С рандомизацией	3	0,66	127	0,47	4	0,49	19
		0,45	8191	0,46	6	0,49	23
	4	0,45	8192	0,65	8	0,67	19
		0,65	8191	0,59	16	0,67	23
	5	0,65	8184	0,62	10,13	0,67	21,23
		0,65	8191	0,56	23	0,66	19
	10	0,67	4093	0,67	11,13	0,67	19,21
	15	0,67	4093	0,67	11,13,15	0,68	23

Влияние рандомизации на коэффициент заполнения иллюстрируется табл.8. Параметры эксперимента (тип текста; α и т.д.) те же, что и в п.6.3. Используется стандартная кодировка символов.

Анализ таблицы показывает, что рандомизация так же, как и перекодировка, целесообразна в тех случаях, когда наблюдаются существенные отклонения в худшую сторону от пуассоновских оценок (см. вывод 2) из п.6.3). Рандомизация в значительной степени устраняет аномальные эффекты в методах деления (при $N = 2^m \pm k$; $k = 0,1$; $l = 3,4,5$) и середины квадрата ($l = 5$, УЦ), а также стабилизирует и улучшает результаты в методах свертки и умножения на обратное число.

Заметим, что процедуры перекодировка и рандомизации, хотя и выполняют сходные функции, но все же не полностью дублируют друг друга. Поэтому последовательное их применение может привести к улучшению результата. К примеру, переход от варианта ($l = 5$, $\Delta = 13$, кодировка типа 0, рандомизация) к вариантам с теми же параметрами, но с кодировками типа 1 (или 2) позволяет повысить (соответственно понизить) коэффициент заполнения с 0,62 до 0,66 (0,53).

7. Выводы и рекомендации.

1. Схема формирования адресов, реализуемая методами деления и середины квадрата с оптимально подобранными параметрами, хорошо описывается пуассоновской моделью для класса ключей, представимых 1-грамматикой связного текста. Метод свертки сопоставим по результатам с двумя первыми методами лишь при кратности наложений, не меньшей трех.

2. При $N \neq 2^m$ целесообразно использовать метод деления. К отклонениям от пуассоновской модели может привести выбор в качестве делителя: а) четных N ; б) N , кратных трем, при значности кода символов исходного алфавита, равной 8; в) $N = 2^m \pm k$, где k – малые целые положительные числа. При длине ключа, меньшей размеров машинного слова, ключ желательно размещать в младших разрядах. Для уменьшения времени счета может быть использован табличный алгоритм вычисления остатка.

3. При $N = 2^m$ и длинных ключах ($l \geq 10$) целесообразно использовать метод свертки, обеспечивающий в этой ситуации результаты, близкие к пуассоновским оценкам, при минимальных (по сравнению с двумя другими методами) временных затратах.

4. При $N = 2^n$ и коротких ключах ($l < 10$) целесообразно использовать метод середины квадрата (УФ). Рекомендуется размещать ключ в старших разрядах машинного слова и выделять под адрес точно средние или, если это невозможно, ближайшие к ним старшие разряды произведения. Временные затраты на реализацию метода существенно меньше чем в случае деления.

5. В случае избыточности кодировки символов исходного алфавита или несогласованности ее с частотой встречаемости символов в массиве ключей рекомендуется целенаправленная перекодировка символов алфавита. Данный прием способствует устранению многих аномальных эффектов, особенно в методах деления и свертки.

6. Использование суперпозиции метода свертки в ее простейшем варианте с исследуемым методом может служить хорошим "профилактическим" средством при работе с плохо изученным классом ключей или новой функцией расстановки.

Литература

1. ВЕЛИЧКО В.М., ГУСЕВ В.Д., КОСАРЕВ Ю.Г., ЛОЗОВСКИЙ В.С., ТИТКОВА Т.Н. Ассоциативное кодирование: реализация и применение. -В кн.: Вычислительные системы. Вып. 62. Ассоциативное кодирование. Новосибирск, 1975, с. 3-37.

2. LUM V.Y., YUEN P.S.T., DODD M.M. Key-to-address transform techniques: a fundamental performance study on large existing formatted files.-"Communs ACM", 1971, v.14, N 4, p.228-239.

3. SEVERANCE D., DUHNE R. A practitioner's guide to address algorithms.-"Communs ACM", 1976, v. 19, N 6, p.314-326.

4. ГУСЕВ В.Д., КОСАРЕВ Ю.Г., ТИТКОВА Т.Н. О задаче поиска повторяющихся отрезков текста. -В кн.: Вычислительные системы. Вып. 62. Ассоциативное кодирование. Новосибирск, 1975, с.49-71.

5. ТИТКОВА Т.Н. Табличный алгоритм вычисления остатков от деления произвольных целых чисел на фиксированное целое число. -Настоящий сборник, с. 102-106.

6. KNUTH D.E. The art of programming. Vol.3. Sorting and searching. 1972. Addison Wesley. Publishing Company.

Поступила в ред.-изд.отд.
18 апреля 1978 года