

Рациональное заполнение перемешанной таблицы

С. З. Свердлов, Д. С. Тропина

Исследована длина поиска в перемешанной таблице с цепочками. Сформулированы рекомендации по рациональному выбору проектного значения степени заполнения, исходя из равенства средней длины удачного и неудачного поиска и приемлемой доли пустых цепочек.

При создании информационных систем широко используются перемешанные таблицы (называемые иногда хеш-таблицами), обеспечивающие быстрый поиск данных по ключу [1]. Одной из простых и эффективных разновидностей таких таблиц являются перемешанные таблицы с цепочками. При проектировании систем, в которых используется хеш-поиск, возникает вопрос о выборе размера такой таблицы. Увеличение размера ускоряет поиск, но требует дополнительного расхода памяти. В этой работе обосновываются рекомендации по выбору разумной степени заполнения и размера перемешанной таблицы с цепочками.

Рассмотрим таблицу, организованную в виде массива указателей на записи, содержащие ключ (поле, по которому выполняется поиск) и данные, сопоставленные данному ключу. Ниже приведено описание на языке Паскаль типов данных для такой таблицы.

```
tPtr = ^tItem; {Указатель на элемент таблицы}
tItem = record {Тип отдельной записи таблицы}
  key: tKey; {Ключ}
  data: tData; {Данные}
  next: tPtr {Указатель на следующую запись}
end;
tHash = array [0..n-1] of tPtr; {Таблица (основной массив)}
```

Здесь n — размер таблицы (ее основного массива). Тип ключа ($tKey$) и тип данных ($tData$) зависят от конкретной задачи. Предполагается, что тип $tKey$ допускает сравнения на равенство.

Схематически организация перемешанной таблицы с цепочками показана на рис. 1.

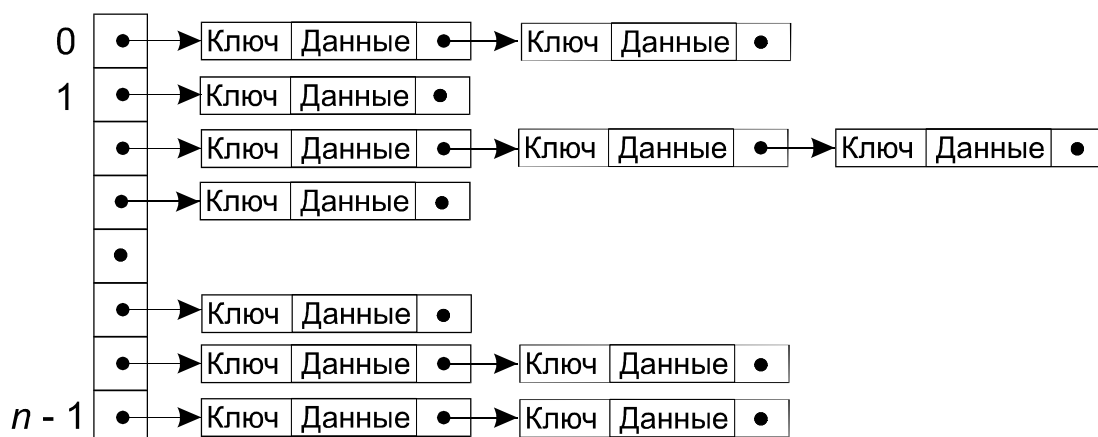


Рис. 1. Схема перемешанной таблицы с цепочками

Заполнение перемешанной таблицы данными происходит следующим образом. Данные со своими ключами заносятся в таблицу последовательно. При добавлении в таблицу пары ключ–данные вычисляется значение функции расстановки для заданного ключа K . Функция расстановки $h(K)$ отображает множество ключей в множество целых из диапазона $0 \dots n - 1$. Значение $h(K)$ используется как индекс в массиве перемешанной таблицы. Пара ключ–значение добавляется в цепочку с этим индексом. Предполагается, что не может быть совпадающих ключей. Хорошая функция расстановки распределяет ключи по таблице возможно более равномерно.

Поиск в таблице заданного ключа выполняется подобно занесению этого ключа в таблицу. Процедура поиска представлена ниже.

```

procedure Search(var Hash: tHash; K: tKey; var Result: tPtr);
var
    p: tPtr;
begin
    p := Hash[h(K)];
    while (p<>nil) and (p^.key<>K) do
        p := p^.next;
    Result := p;
end;

```

Результат поиска — указатель на элемент таблицы с искомым ключом, если ключ найден (удачный поиск) или значение **nil**, если искомый ключ отсутствует в таблице (неудачный поиск).

Оценим длину удачного и неудачного поисков в перемешанной таблице с цепочками. Длиной поиска назовем количество элементов цепочки, просматриваемых в ходе поиска, или, что в данном случае то же самое, количество выполняемых при поиске сравнений ключей.

Неудачный поиск. Предположим, что расстановка обеспечивает на множестве ключей, отсутствующих в таблице, равномерное распределение значений $h(K)$ в диапазоне $0 \dots n-1$. При отсутствии данных о конкретной задаче, при решении которой используется таблица, такое предположение представляется вполне естественным. При неудачном поиске происходит сравнение искомого ключа K с ключами всех элементов цепочки с индексом $h(K)$, то есть длина отдельного поиска равна длине соответствующей цепочки. Если цепочка пуста, длина поиска будет нулевой. Поскольку вероятность выбора цепочек одинакова, средняя длина поиска будет равна средней длине цепочки, которая в свою очередь равна m/n , где n — размер массива перемешанной таблицы; m — количество занесенных в таблицу пар ключ–данные.

Итак, средняя длина неудачного поиска

$$D_{cp.}^{неуд.} = \frac{m}{n} = \sigma . \quad (1)$$

Здесь $\sigma = \frac{m}{n}$ — степень заполнения таблицы. Заметим, что степень заполнения σ для таблицы с цепочками может быть больше 1.

Удачный поиск. Поиск удачен, если запись с искомым ключом имеется в таблице. Считая, что при удачном поиске функция расстановки с равной вероятностью выбирает одну из непустых цепочек, можно заключить, что средняя длина удачного поиска определяется средней длиной непустой цепочки (пустые цепочки при поиске заведомо имеющегося в таблице ключа не могут быть выбраны). При линейном поиске в цепочке длины L число сравнений в среднем будет равно $(L+1)/2$, поэтому

$$D_{cp.}^{удач.} = \frac{L_{cp.}^{непуст.} + 1}{2} , \quad (2)$$

где $L_{cp.}^{непуст.}$ — средняя длина непустой цепочки.

Определим среднюю длину непустой цепочки. Обозначим E_m — среднее число пустых цепочек в таблице с m записями. Тогда

$$L_{cp.}^{непуст.} = \frac{m}{n - E_m}. \quad (3)$$

Если $m=0$ (таблица пуста), $E_m = n$ — все цепочки пусты. Если в таблицу с m записями и E_m пустыми цепочками добавляется очередной элемент, то $E_{m+1} = p_{E_m} E_m + p_{E_m-1} (E_m - 1)$, где p_{E_m} — вероятность того, что после добавления очередного элемента число пустых цепочек не изменится (новый элемент будет добавлен к существующей цепочке); p_{E_m-1} — вероятность уменьшения числа пустых цепочек.

С учетом того, что $p_{E_m} = \frac{n - E_m}{n}$; $p_{E_m-1} = \frac{E_m}{n}$, получаем рекуррентную формулу: $E_{m+1} = \frac{n - E_m}{n} E_m + \frac{E_m}{n} (E_m - 1) = E_m \frac{(n - 1)}{n}$. Поскольку $E_0 = n$, в явном виде среднее количество пустых цепочек в таблице с m записями выражается формулой

$$E_m = n \left(\frac{n - 1}{n} \right)^m = n \left(1 - \frac{1}{n} \right)^m.$$

Поскольку $\lim_{n \rightarrow \infty} \left(n - \frac{1}{n} \right)^n = e^{-1}$, при больших n :

$$E_m \approx n e^{-\sigma}. \quad (4)$$

В реальных приложениях обычно $n > 10$. В этих условиях полученное приближенное равенство выполняется с высокой точностью. Подставляя (4) в (3), а (3) в (2), получаем:

$$D_{cp.}^{удач.} \approx \frac{\sigma}{2(1 - e^{-\sigma})} + \frac{1}{2}. \quad (5)$$

На рис. 2 показаны графики зависимости средней длины удачного и неудачного поисков от степени заполнения перемешанной таблицы с цепочками.

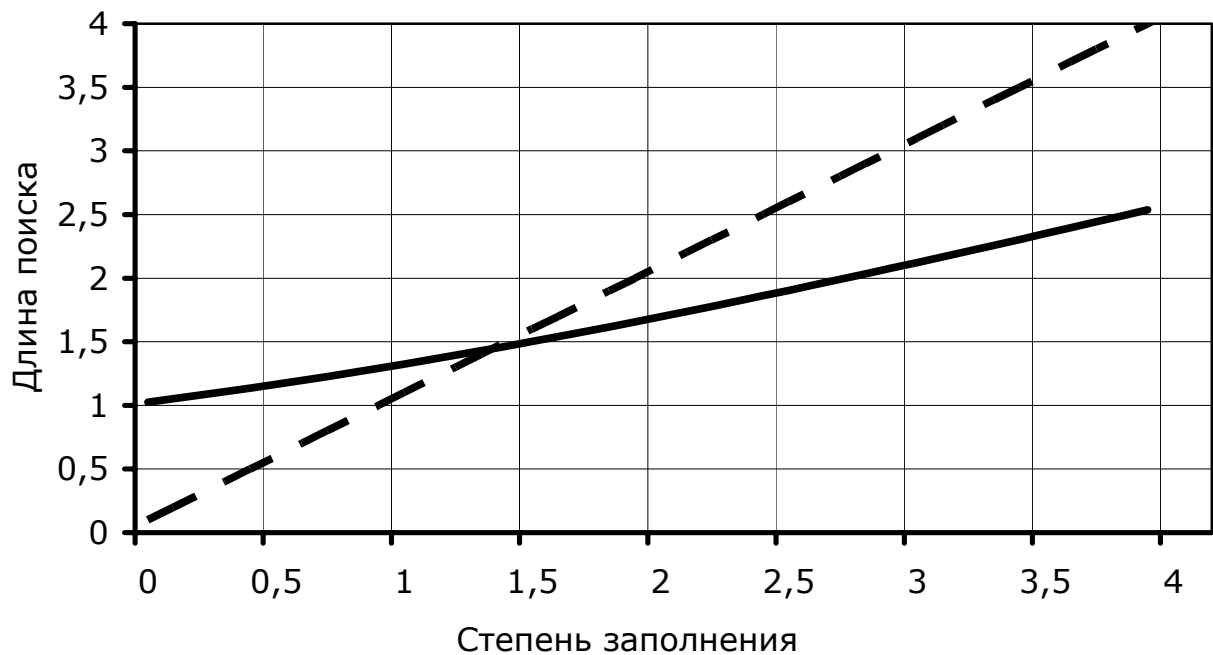


Рис. 2. Зависимость средней длины поиска от степени заполнения. Сплошная линия — удачный поиск; пунктирная линия — неудачный.

При проектировании информационной системы, использующей перемешанную таблицу, возникает вопрос о выборе размера основного массива таблицы (n) и степени заполнения (σ). Нерациональным является как выбор слишком маленького проектного значения σ — объем основного массива оказывается избыточным, так и выбор слишком большого значения σ — увеличивается длина поиска. Какое же значение степени заполнения можно считать разумным компромиссом? Один из возможных ответов можно получить из рассмотрения графика на рис. 2. В качестве оптимального значения степени заполнения предлагается выбрать значение σ^* , при котором пересекаются линии удачного и неудачного поиска на графике. При такой степени заполнения средняя длина удачного и неудачного поиска будут равны. Значение σ^* можно вычислить, приравняв правые части формул (1) и (5) и решив полученное уравнение. Численное решение дает $\sigma^* \approx 1,4455$.

При равенстве средней длины удачного и неудачного поисков среднее время поиска не будет зависеть от доли удачных и неудачных поисков при работе информационной системы. Это облегчает её проектирование. Основной

массив таблицы используется при этом достаточно эффективно, доля пустых цепочек, вычисленная с помощью формулы (4), будет составлять в среднем 23,6%. Средняя длина поиска будет равна 1,4455. Поиск с такой средней длиной (для любых, в том числе и больших значений n) можно оценить как очень эффективный.

Усреднение по всем удачным поискам. Проведенные выше выкладки, приведшие к формуле (5) для средней длины удачного поиска, выполнены в предположении о равновероятном выборе непустой цепочки при поиске. Однако такое предположение может выполняться не всегда.

Когда речь идет об удачном поиске в таблице с m записями, усреднение можно выполнить по m возможным удачным поискам (поиск каждого из m ключей выполняется один раз). Частота выбора цепочки в этом случае будет равна количеству записей в этой цепочке (длине цепочки).

Опуская громоздкие промежуточные выкладки, приведем формулу для средней длины поиска в этом случае:

$$D_{cp.}^{удач.} = 1 + \frac{m-1}{2n} \quad (6)$$

При больших n и m можно использовать приближенную формулу:

$$D_{cp.}^{удач.} \approx 1 + \frac{\sigma}{2} \quad (7)$$

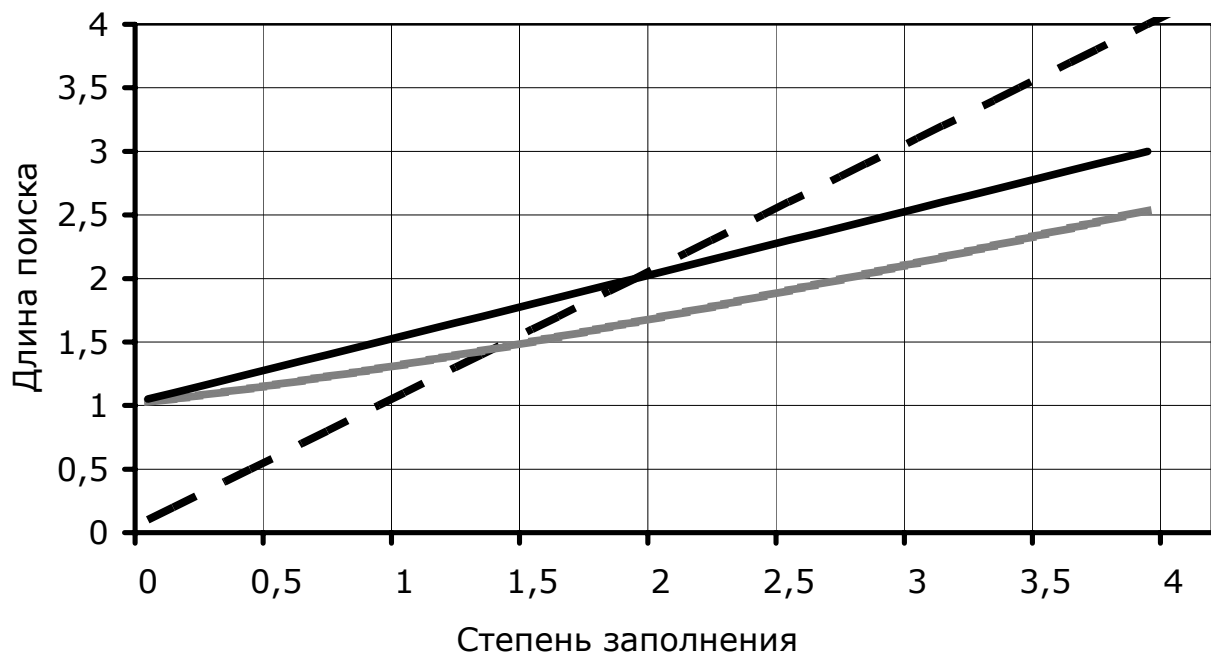


Рис. 3. Зависимость средней длины поиска от степени заполнения перемешанной таблицы.

Сплошные линии — удачный поиск; пунктирная линия — неудачный.

На рис. 3 показаны графики зависимости длины поиска от степени заполнения таблицы, рассчитанные по формулам (1), (5) и (7). Приравнивая правые части (1) и (7), получаем, что средняя длина удачного поиска, вычисленная усреднением всех удачных поисков, и средняя длина неудачного поиска оказываются равны при $\sigma^{**}=2$.

Уточняя сформулированные выше соображения по выбору рациональной степени заполнения перемешанной таблицы с цепочками, можно заключить, что проектные значения степени заполнения, обеспечивающие примерное равенство длины удачного и неудачного поисков, можно выбирать в диапазоне от 1,45 до 2,0. Выбор большего значения позволяет уменьшить потери памяти из-за присутствия пустых цепочек в основном массиве. Если расход памяти критичен, можно увеличить степень заполнения и выше указанных значений, удлинив поиск (в большей степени неудачный, в меньшей — удачный). Подходящее значение степени заполнения можно вычислять, исходя из приемлемой доли пустых цепочек. Обозначая эту долю $\varepsilon = \frac{E_m}{n}$, из (4) получаем:

$$\sigma = -\ln \varepsilon. \quad (8)$$

Например, при доле пустых цепочек $\varepsilon=0,1$ (10%), получаем $\sigma \approx 2,3$.

Можно также руководствоваться данными табл. 1.

Табл. 1. Длина поиска и процент пустых цепочек в перемешанной таблице

Степень заполнения	Длина удачного поиска	Длина неудачного поиска	Процент пустых цепочек
1,00	1,29...1,50	1,00	36,8
1,50	1,47...1,75	1,50	22,3
2,00	1,66...2,00	2,00	13,5
2,50	1,86...2,25	2,50	8,2
3,00	2,08...2,50	3,00	5,0
3,50	2,30...2,75	3,50	3,0
4,00	2,54...3,00	4,00	1,8
4,50	2,78...3,25	4,50	1,1

Литература

1. Лебедев В. Н. Введение в системы программирования. М., «Статистика», 1975. 312 с. ил.

Сведения об авторах

1. Свердлов Сергей Залманович, 1954 г.р., к.т.н., доцент. Область научных интересов: программирование, компьютерные технологии, языки программирования и их реализация, цифровая фотография и обработка изображений. Тел. +7 921-122-74-43
2. Тропина Дарья Сергеевна, 1983 г.р., старший аналитик. Область научных интересов: информационные системы.