УДК 681.3

О. В. Герман, канд. техн. наук, доцент; Н. Н. Дорожкина,
канд. физ.-мат. наук; А. Р. Самко, аспирант

ПОИСК ИНФОРМАЦИИ НА ОСНОВЕ КОЛЛЕКЦИИ B-ДЕРЕВЬЕВ

A new kind of addressing tools is introduced which combines capabilities of associative memory, B-trees and pattern recognition mechanism. The proposed technique enables one to obtain the address of the retrieved data in a more efficient way then that one based on usage of indexing files. Each digit of the address value is calculated on the basis of B-tree which consists of a single node at the best and of the number of square root of N in average (N stands for the number of nodes). The suggested approach admits efficient concurrency of address estimation procedure and more evenly uses memory space in comparison with associative search. It may be used in the recognition systems as well as an alternative to neuro-nets because it doesn't need learning procedure and provides more compact classifying search trees.

Введение. В настоящее время используют следующие основные механизмы ускорения поиска информации: B-деревья, ассоциативный поиск, основанный на применении функции хеширования, которая преобразует признаки искомых данных непосредственно в физические адреса, инвертированные файлы, а также механизмы поиска по ключевым словам, широко используемые в сети Интернет [1–3]. У каждого из указанных механизмов есть свои достоинства и недостатки. Основная задача поиска такова – быстро получить адрес искомых данных. Различие между рассматриваемыми механизмами состоит в следующем:

1) объеме используемой вспомогательной информации;

2) возможности распараллеливания процессов вычисления адреса;

3) наличии конфликтных ситуаций, когда различные данные адресуются по одинаковым адресам;

4) скорости получения адресов.

Наиболее важные характеристики системы поиска, несомненно, – это обеспечиваемая скорость поиска информации и используемый объем вспомогательной информации. Предлагаемая статья содержит новый подход к реализации поиска. В результате поиска определяется физический адрес многомерных данных. Каждый разряд адреса вычисляется на основе дерева, которое в лучшем случае состоит из единственной вершины, а в среднем содержит число вершин, которое определяется по результатам экспериментов как корень квадратный от числа всех записей в базе данных (в B-дереве число всех вершин равно числу всех записей). Таким образом, затраты на реализацию вспомогательной памяти в предлагаемом методе в среднем оцениваются как $N^{0,5}(\log_2 N)$, в то время как у B-дерева эта оценка составляет N. В сравнении с ассоциативным поиском предлагаемый механизм не порождает конфликтов и равномерно использует все адресное пространство.

Описание механизма поиска. Пусть имеется два числовых признака x_1 и x_2 и множество значений, собранных в табл. 1.

Таблица 1

x_1	2	–1	0	2	3	6	1	4
x_2	4	3	1	5	–2	3	1	3

Разобьем общее число записей на две равные (примерно равные) половины, из которых одну обозначим как «0», а вторую – как «1». Затем также поступим с полученными половинами и т. д. Результат такого разбиения представлен на рис. 1.

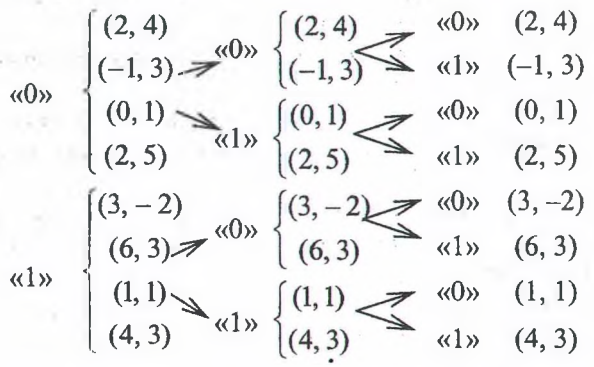

Рис. 1. Определение адресов наборов значений

Теперь допустим, что для разбиения каждого множества на две равные (примерно равные) половины мы используем линейное неравенство вида

$$a_1 x_1 + a_2 x_2 \geq 0. \qquad (1)$$

Коэффициенты a_1 и a_2 нам еще предстоит найти. Использование неравенства (1) состоит в том, что вместо x_1 и x_2 мы подставляем значения, соответствующие отыскиваемому объекту из табл. 1. Если неравенство выполняется, то попадаем в половину, помеченную как «0». Если неравенство не выполняется, то попадаем в половину, обозначенную как «1». Этот принцип можно эффективно обобщить, так что в лучшем случае число всех неравенств будет равно $1+\lfloor\log_2 N\rfloor$, где N – число всех записей. Однако эта наилучшая теоретическая оценка, к сожалению, не всегда выдерживается.

Покажем, как отыскать коэффициенты a_1 и a_2 в (1). Составим табл. 2 для первых двух множеств.

Таблица 2

x_1	2	–1	0	2	3	6	1	4
x_2	4	3	1	5	–2	3	1	3
y	≥ 0				< 0			

Запишем следующую систему неравенств:

$$\begin{aligned} 2a_1 + 4a_2 &\geq 0, \\ -1a_1 + 3a_2 &\geq 0, \\ 0a_1 + 1a_2 &\geq 0, \\ 2a_1 + 5a_2 &\geq 0, \\ 3a_1 - 2a_2 &< 0, \\ 6a_1 + 3a_2 &< 0, \\ 1a_1 + 1a_2 &< 0, \\ 4a_1 + 3a_2 &< 0. \end{aligned} \qquad (2)$$

Неравенства (2) составлены согласно описанному принципу. Если найти коэффициенты a_1 и a_2, то все множество записей будет разделено нужным нам образом на две условные половины: «0» и «1». Имеются две проблемы. Во-первых, следует избавиться от жестких неравенств (<). Это сделать нетрудно, если ввести достаточно малую величину $\xi > 0$, такую что

$$c_1a_1 + c_2a_2 < 0$$

можно заменить на

$$c_1a_1 + c_2a_2 \leq -\xi.$$

Заметим, что такая замена может сделать систему неравенств несовместной, однако и исходная система может быть несовместна, так что наша основная задача – разрешить проблему несовместности системы неравенств. В иллюстративных целях зададим $\xi = 1$ и приведем все неравенства к виду $\geq$:

$$\begin{aligned} 2a_1 + 4a_2 &\geq 0, \\ -1a_1 + 3a_2 &\geq 0, \\ 0a_1 + 1a_2 &\geq 0, \\ 2a_1 + 5a_2 &\geq 0, \\ -3a_1 + 2a_2 &\geq 1, \\ -6a_1 - 3a_2 &\geq 1, \\ -1a_1 - 1a_2 &\geq 1, \\ -4a_1 - 3a_2 &\geq 1. \end{aligned} \qquad (3)$$

Для решения мы используем алгоритм устранения невязок [4].

Определение. Неравенство с положительной правой частью называется невязкой.

Если в системе нет невязок, то ее решение доставляется нулевыми значениями переменных. В противном случае описываемый алгоритм пытается избавиться от невязок. При этом выделяются две фазы. На первой фазе нужно получить систему базисных неравенств вида $a_i \geq 0$. Вторая фаза выполняется так же, как и первая, но уже при наличии базисных неравенств. Если на второй фазе в процессе итераций встречается невязка, причем все коэффициенты в левой части неположительны, то устанавливаем факт неразрешимости (несовместности) системы неравенств. Этот факт мы будем использовать специальным образом. Итак, возьмем любую невязку, например,

$$-3a_1 + 2a_2 \geq 1$$

и выразим из нее переменную a_1 так

$$-3a_1 \geq 1 + 2a_2,$$

$$a_1 = -\frac{1}{3} - \frac{2}{3}a_2 - z_1.$$

Здесь z_1 – новая неотрицательная переменная. Подставим вместо a_1 полученное выражение. Получим следующую систему:

$$\begin{aligned} \frac{8}{3}a_2 - 2z_1 &\geq \frac{2}{3}, \\ \frac{11}{3}a_2 + z_1 &\geq -\frac{1}{3}, \\ a_2 &\geq 0, \\ \frac{11}{3}a_2 - 2z_1 &\geq \frac{2}{3}, \\ z_1 &\geq 0, \\ a_2 + 6z_1 &\geq -1, \\ -\frac{1}{3}a_2 + z_1 &\geq \frac{2}{3}, \\ -\frac{1}{3}a_2 + 4z_1 &\geq -\frac{1}{3}. \end{aligned}$$

Первая фаза завершена. Имеются два базисных неравенства $a_2 \geq 0$ и $z_1 \geq 0$. Вторая фаза выполняется с небольшим отличием от первой: из невязки выражаем переменную с положительным коэффициентом. Так, возьмем невязку

$$a_2 \geq \frac{1}{4} + \frac{3}{4}z_1.$$

Выражаем a_2:

$$a_2 = \frac{1}{4} + \frac{3}{4}z_1 + z_2.$$

Выполняя подстановки подобным образом, получим систему без невязок. В этой системе решение дает $z_2 = z_3 = 0$. Отсюда в силу произведенных подстановок найдем: $a_2 = -2$, $a_2 = 1$.

Итак, первое разделяющее неравенство нами получено в форме

$$-2x_1 + x_2 \geq 0.$$

Нетрудно убедиться, что для точек (2, 4), (–1, 3), (0, 1), (2, 5) мы получили условное множество «0» (неравенство (1) выполнимо). Для остальных точек это неравенство не выполнимо (условное множество «1»).

Теперь, согласно рис. 1, нам нужно провести новую разбивку точек в соответствии с табл. 3.

Таблица 3

x_1	2	–1	0	2	3	6	1	4
x_2	4	3	1	5	–2	3	1	3
y	≥ 0		< 0		≥ 0		< 0	

Действительно, это второй слой точек на рис. 1 разбит именно таким образом. Снова составляем систему неравенств (для переменных a_3 и a_4):

$$2a_3 + 4a_4 \geq 0,$$
$$-1a_3 + 3a_4 \geq 0,$$
$$0a_3 - a_4 \geq 1,$$
$$-2a_3 - 5a_4 \geq 1,$$
$$3a_3 - 2a_4 \geq 0,$$
$$6a_3 + 3a_4 \geq 0,$$
$$-a_3 - a_4 \geq 1,$$
$$-4a_3 - 3a_4 \geq 1.$$

Выполняем первую фазу алгоритма. Из невязки

$$0a_3 - a_4 \geq 1$$

получим

$$0a_3 - a_4 = 1 + z_1,$$
$$a_4 = -1 - z_1.$$

Эта замена приводит к следующей системе:

$$2a_3 - 4z_1 \geq 4,$$
$$-a_3 - 3z_1 \geq 3,$$
$$z_1 \geq 0,$$
$$-2a_3 + 5z_1 \geq -4,$$
$$3a_3 + 2z_1 \geq -2,$$
$$6a_3 - 3z_1 \geq 3,$$
$$-a_3 + z_1 \geq 0,$$
$$-4a_3 + 3z_1 \geq -2.$$

Из невязки

$$2a_3 - 4z_1 \geq 4$$

выразим переменную a_3:

$$a_3 = 2 + 2z_1 + z_2.$$

Теперь система (3) примет следующий вид:

$$\text{(a)}\ z_2 \geq 0, \qquad (4)$$
$$\text{(b)}\ -5z_1 - z_2 \geq 5,$$
$$\text{(c)}\ z_1 \geq 0,$$
$$\text{(d)}\ z_1 - 2z_2 \geq 0,$$
$$\text{(e)}\ 8z_1 + 3z_2 \geq -8,$$
$$\text{(f)}\ 9z_1 + 6z_2 \geq -9,$$
$$\text{(g)}\ -z_1 - z_2 \geq 2,$$
$$\text{(h)}\ -5z_1 - 4z_2 \geq 6.$$

Теперь мы столкнулись со следующей ситуацией. Невязки (b), (g), (h) не выполнимы вместе с базисными неравенствами. Следовательно, система противоречива. Это означает, что нельзя подобрать переменные a_3, a_4 в (4), чтобы разрешить нужным образом записи на подмножества «0» и «1». Теперь поступим так: просто исключим из (4) невязки (b), (g), (h), которые с базисными неравенствами дают противоречие.

В итоге не останется ни одной невязки и процесс завершится. Найдем

$$z_1 = z_2 = 0,\ \ a_3 = 2,\ \ a_4 = -1$$

и мы получим снова разделяющее неравенство

$$2x_1 - x_2 \geq 0. \qquad (5)$$

Однако одним неравенством (5) нам обойтись не удастся, так как оно неверно разделяет точки, соответствующие исключенным неравенствам.

Сначала сформируем следующую систему:

$$2a_5 + 4a_6 \geq 0,$$
$$3a_5 - 2a_6 \geq 0,$$
$$6a_5 + 3a_6 \geq 0, \qquad (6)$$
$$-1a_5 - 1a_6 \geq 1,$$
$$-4a_5 - 3a_6 \geq 1.$$

Эта система составлена для тех точек, для которых неравенство (5) выполняется – условное множество «0». Однако в это множество неверно отнесены точки (1, 1) и (4, 3).

Опуская выкладки, запишем решение системы: $a_5 = 1$, $a_6 = -2$ и новое разделяющее неравенство

$$x_1 - 2x_2 \geq 0.$$

При поиске решения (6) пришлось удалить одну из точек, поэтому процесс построения разделяющих неравенств следует продолжить.

Приведенное подробное описание отменяет необходимость объяснять последующие шаги. В результате для среднего слоя точек на рис. 1 будет построено дерево, приведенное на рис. 2, узлы которого соответствуют разделяющим неравенствам. Теперь должно быть ясно, как выполняется формирование второго разряда адреса с использованием поискового дерева, показанного на рис. 2.

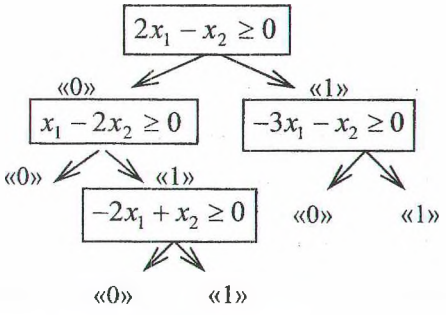

Рис. 2. B-дерево для второго разряда адреса

Например, входим в корневую вершину с набором (2, 5). Первое неравенство не выполняется, следовательно, переходим по ветке «1» в узел

$$-3x_1 - x_2 \geq 0.$$

В этом узле неравенство также не выполняется. Окончательно определяем результат поиска «1».

Из подобных же соображений строится дерево поиска для третьего разряда адреса (рис. 3).

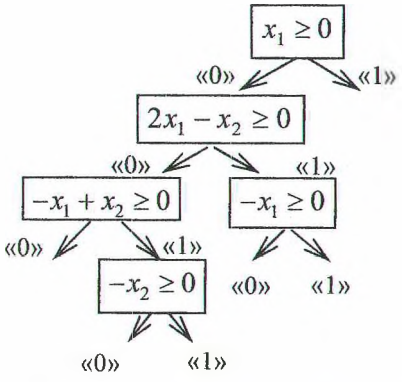

Рис. 3. B-дерево для третьего разряда адреса

Читатель, вероятно, догадался, почему мы употребили словосочетание «поисковое дерево». В самом деле, движение по ветвям такого дерева вполне соответствует процессу поиска на B-дереве. Обратим также внимание на аналогию в данном случае поиска и распознавания. Так, с помощью поискового дерева типа изображенного на рис. 2 и 3 можно ответить на вопрос, к какому классу («0» или «1») принадлежит заданная точка. Этот вопрос является основным в теории распознавания образов. Отсюда мы и почерпнули аналогию, которую закрепили в названии.

Заключение. Основные достоинства разработанного механизма поиска можно резюмировать следующим образом:

1) формирование разрядов адреса может осуществляться параллельно;

2) поиск можно при необходимости реализовать как распознавание;

3) затраты памяти на представление поисковых деревьев в среднем даже ниже, чем в случае B-дерева.

Литература

1. Кричевский, Р. Е. Сжатие и поиск информации / Р. Е. Кричевский. – М.: Радио и связь, 1989. – 166 с.

2. Бауэр, Ф. Информатика / Ф. Бауэр, Г. Гооз. – М.: Мир, 1976. – 486 с.

3. Джордж, Ф. Основы кибернетики / Ф. Джордж. – М.: Радио и связь, 1984. – 272 с.

4. Герман, О. В. Различные приложения стратегии устранения невязок / О. В. Герман, Н. Н. Дорожкина // Вестник Ставропольского государственного университета. – 1999. – № 20. – С. 85–99.