УДК 518.5:007:621.391:519.2

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ЗАДАЧИ ТАКСОНОМИИ

Ю.Г. Косарев, Н.В.Кучин

Для решения задачи таксономии В.Н.Ёлкиной и Н.Г.Загоруйко [1, 2] предложен эвристический алгоритм Форэль, зарекомендовавший себя на практике. Идея алгоритма заключается в следующем.

Задано множество точек в многомерном пространстве. Выбирается радиус R гиперсферы. Центр гиперсферы совмещается с одной из точек. Определяется центр тяжести (ЦТ) для всех точек, попавших внутрь гиперсферы. Центр гиперсферы совмещается с ЦТ. Находится новый ЦТ и т.д., до тех пор пока гиперсфера не остановится. Тогда все точки, находящиеся внутри гиперсферы, объявляются таксоном. Для оставшихся точек повторяется тот же процесс до выявления всех таксонов. Затем задача решается для другого значения R и т.д.

Особенность данного алгоритма заключается в том, что все координаты точек приходится хранить в оперативной памяти. Удачного решения использования для этой цели вспомогательных памятей пока нет, что ограничивает объем реализуемых задач таксономии. Кроме того, при увеличении объема заметно возрастает и время счета.

Для преодоления указанных трудностей в данной работе

предлагается использовать параллельный счет на нескольких машинах, объединенных в систему типа "Минск-222" [3], упростить счет путем замены гиперсферы гиперкубом, применить более плотную и в то же время удобную для счета форму расположения исходной информации и параллельно обрабатывать много точек (в одной машине) с помощью логических операций.

I. Замена гиперсферы гиперкубом. Основное время счета по алгоритму Фореля приходится на определение расстояния точек от центра гиперсферы. Этого можно избежать, если заменить гиперсферу гиперкубом с ребрами, параллельными осям координат. В этом случае попадание точки внутрь куба определяется независимо по каждой координате. Счет точки можно прекращать на первой же координате, оказавшейся вне куба. Открываются также другие возможности сокращения счета, о чем будет сказано ниже.

Замена гиперсферы гиперкубом не должна оказать существенного влияния на качество таксономии и находится в согласии с введенным Н.Г.Загоруйко критерием информативности [4]. В соответствии с этим критерием в качестве решающих функций, как правило, лучше применять гиперплоскости, а не сложные гиперповерхности.

2. Предварительное упорядочение точек. Замена сферы кубом делает целесообразным предварительное упорядочение точек по одной из координат. Тогда все точки, попадающие внутрь куба по этой координате, оказываются собранными вместе. Их легко выделить и тем самым исключить из рассмотрения те точки, которые заведомо в куб не попадают.

3. Вертикальная запись координат точек. Разрядность координат точек, как правило, распределена в небольшом интервале, чаще всего от I до 10 бит. Это затрудняет плотное и одновременное размещение в памяти исходных данных. Объединение нескольких координат в одно слово усложняет выделение каждой из них в процессе счета. Единообразное размещение обычно наносит ущерб компактности.

Чтобы избежать этих трудностей, предлагается разместить информацию по вертикали, т.е. отвести для координаты I-й точки I-е разряды подряд идущих слов памяти, для 2-й – 2-е разряды и

т.д. (рис. I). Таким образом, каждое m -разрядное слово содержит по одному разряду от m точек.

При размещении координаты точек упорядочиваются по возрастанию их разрядности. Это позволяет описать размещение исходного массива с помощью списка длиной n_{max} – наибольшей разрядности координат.

4. Параллельный счет при выделении точек, попавших внутрь гиперкуба. Вертикальная запись и замена сферы кубом позволяют одновременно вести счет до m точек с помощью логических операций.

Пусть $A_j^o(a_1,\dots,a_n)$ и $B_j^o(b_1,\dots,b_n)$ – нижняя и верхняя границы гиперкуба по координате j, $X_{ji}(x_{1i},\dots,x_{ni})$ – j -я координата i -й точки, где a_κ, b_κ, $x_{\kappa i} \in \{0,1\}$, n – разрядность координаты j.

$$A_j^o(a_1,\dots,a_n) = a_1\cdot 2^{n-1}+\dots+a_n\cdot 2^0;$$

$$B_j^o(b_1,\dots,b_n) = b_1\cdot 2^{n-1}+\dots+b_n\cdot 2^0;$$

$$X_{ji}^o(x_{1i},\dots,x_{ni}) = x_{1i}\cdot 2^{n-1}+\dots+x_{ni}\cdot 2^0.$$

При их вертикальной записи образуется n m -разрядных булевых векторов $A_{j\kappa}(a_\kappa,\dots,a_\kappa)$, $B_{j\kappa}(b_\kappa,\dots,b_\kappa)$ и $X_{j\kappa}(x_{\kappa 1},\dots,x_{\kappa m})$, $\kappa = 1,2,\dots,n$.

Введем булевы векторы $Y(y_1,\dots,y_m)$, $U(u_1,\dots,u_m)$ и $V(v_1,\dots,v_m)$ со следующими значениями составляющих:

$y_i = 1$, если i -я точка не исключена из обработки (еще не включена в один из таксонов или не оказалась вне куба по предыдущим координатам);

$u_i = 1$, если для данной точки установлено по предыдущим разрядам данной координаты, что $X_{ji} \geqslant A_j$;

$v_i = 1$, если для данной точки установлено по предыдущим разрядам данной координаты, что $X_{ji} \leqslant B_j$.

Обработку по данной координате можно вести до выполнения для всех m точек условия

$$y_i \wedge (\neg u_i \vee \neg v_i) = 0, \quad i = 1,2,\dots,m. \qquad (I)$$

Выполнение условия

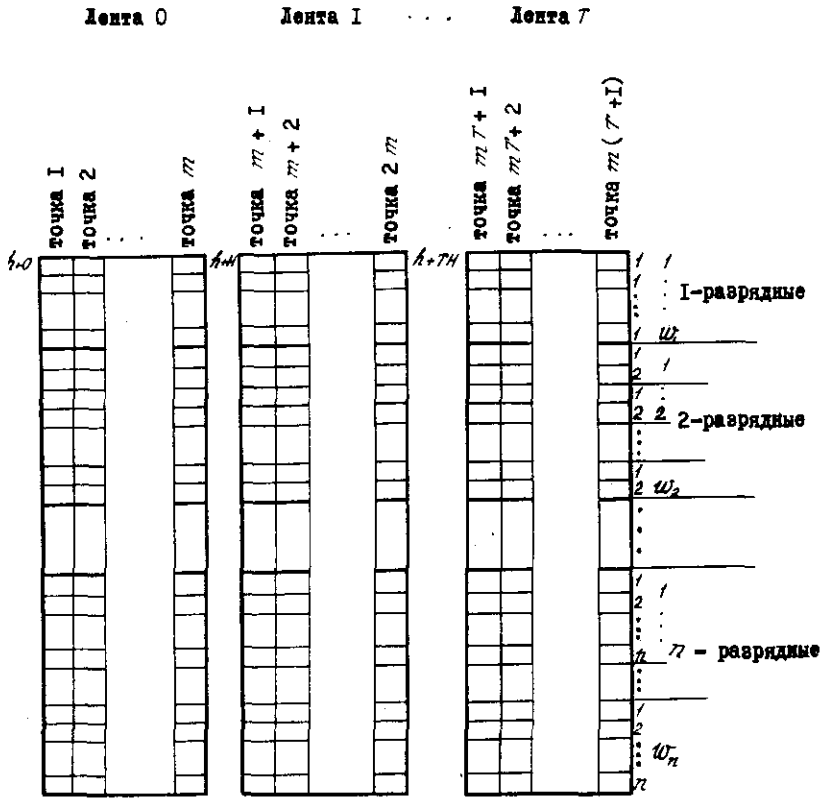

Рис.I. Вертикальное расположение координат точек

$$y_i = 0 \qquad (2)$$

для всех $i = 1, \ldots, m$ означает конец обработки этих точек (все m точек оказались вне куба). Запишем (I) и (2) условно

$$y \wedge (\neg U \vee \neg V) = 0 \ , \qquad (I')$$

$$y = 0 \ . \qquad (2')$$

Процесс сводится к последовательному просмотру векторов X_{ik}, внесению изменений в векторы y, U, V и проверке выполнимости условий (I) и (2).

Удобно рассмотреть каждый из четырех случаев.

а) $a_k = b_k = 0$.

$$U := U \vee (y \wedge X) \ ,$$
$$y := y \wedge (V \vee \neg X) .$$

б) $a_k = 0, b_k = 1$.

$$U := U \vee (y \wedge X) \ ,$$
$$V := V \vee (y \wedge \neg X) .$$

в) $a_k = 1, b_k = 0$.

$$y := \{(U \oplus X) \vee (U \wedge V)\} .$$

г) $a_k = b_k = 1$.

$$V := V \vee (y \wedge \neg X) \ ,$$
$$y := y \wedge (U \vee X) .$$

Здесь символы " $\vee$ ", " $\wedge$ ", " $\oplus$ ", " $\neg$ " означают поразрядные операции дизъюнкции, конъюнкции, сложения по $mod\,2$ и отрицания над m - разрядными булевыми векторами.

5. П а р а л л е л ь н ы й с ч е т п р и о п р е д е л е н и и Ц Т. Вертикальное размещение исходного массива позволяет также сравнительно просто определять ЦТ. Сначала логическим умножением на соответствующий вектор y выделяем точки, попавшие внутрь гиперкуба. Затем с помощью обычно имеющейся в машинах операции суммируем единицы в слове. Накапливаем сумму одноименных разрядов данной координаты. Когда все точки подсчитаны, умножаем полученную сумму на 2, добавляем к результату сумму числа единиц в следующем (младшем) разряде дан-

ной координаты и т.д. по схеме Горнера, пока не получим суммарное значение данной координаты. Затем это значение делится на число точек, попавших в куб, которое определяется суммированием единиц у векторов y .

6. Время счета при вертикальном размещении координат точек зависит от разрядности координат n и числа одновременно считаемых точек τ ($\tau \le m$).

6.1. Программирование для машины "Минск-22" показало, что счет одного разряда у τ точек ($\tau \le 37$) при выделении точек, попавших внутрь куба, занимает почти столько же времени, что и счет координаты одной точки при обычном расположении информации, т.е. можно записать

$$\tau_в = \tau_2 = \tau .$$

Измерим в единицах τ время счета координаты с разрядностью $n = 1,\dots,10$ при числе точек $\tau = 1,\dots,37$. При усреднении по всем возможным 370 комбинациям τ и n получим, что среднее время счета в обоих случаях:

$$t_2' - t_в' = 19\tau - 5,5\tau = 13,5\tau ;$$

$$t_2'/t_в' = 19\tau/5,5\tau \approx 3,5 .$$

В некоторых ситуациях (при $\tau < n$) счет при вертикальном расположении хуже обычного.

6.2. При вычислении ЦТ время обработки одного разряда τ слов ($\tau \le 37$) и время обработки одного значения координаты $n \le 37$ занимает примерно то же время τ, что и при выделении точек, поэтому усредненные соотношения для чистых времен счета будут те же, что и выше. Однако при обычном способе приходится каждый раз проверять, участвует ли данная точка в вычислении ЦТ. Общее число таких проверок (равное произведению числа точек на число координат) составляет в среднем около 9τ . Аналогичная проверка при вертикальной записи выполняется автоматически сразу для 37 точек умножением на вектор y и практически не влияет на время счета.

$$t_2'' - t_в'' = 28\tau - 5,5\tau = 22,5\tau ; \quad t_2''/t_в'' \approx 5,1 .$$

В этом случае при любом соотношении τ и n время счета при вертикальном размещении меньше, чем при обычном.

6.3. Общее среднее время счета одной координаты при вертикальном размещении $\tau_в = 11\tau$, при обычном $\tau_2 = 47\tau$, т.е. в среднем при вертикальном размещении при $n = 1, 2, \dots, 10$ время счета примерно в 4,3 раза меньше обычного.

7. Транспонирование. Преобразование к вертикальной форме записи, если координаты точек размещены в порядке возрастания их разрядности, не занимает много времени.

Разобьем машинные слова на группы по 4 разряда. В "Минске-22" получится 9 групп и один знаковый разряд. Число, заключенное в каждой из этих групп, используем как адрес, по которому обращаемся к переключателю. Каждый из 16 выходов переключателя передает управление соответствующей СП. Эти СП логически добавляют единицы в соответствующие разряды ячеек предварительно очищенного для транспонированного массива поля. Такое транспонирование занимает примерно столько же времени, сколько и счет одного положения гиперкуба, т.е. составляет ничтожную долю от общего времени решения задачи.

8. Параллельный алгоритм. В соответствии с методикой крупноблочного распараллеливания [5] для разделения процесса счета на параллельные ветви могут быть выбраны циклы счета точек и счета координат. Оба эти цикла независимые, охватывают основные операторы и много раз повторяются, т.е. удовлетворяют условиям, необходимым для эффективного распараллеливания. Цикл счета координат стал обладать указанными свойствами лишь после замены гиперсферы гиперкубом.

8.1. При распараллеливании счета точек все они равномерно распределяются между машинами и одновременно обрабатываются. После выделения точек, попавших внутрь гиперсферы, их одноименные координаты суммируются и передаются в другие машины вместе с числом точек внутри гиперсферы. При этом каждая машина определяет свою часть координат ЦТ. Затем машины обмениваются координатами ЦТ, и процесс повторяется.

Серьезный недостаток этого варианта состоит в неравномерном исключении из дальнейшего счета точек из разных машин, что вызывает неодинаковую загрузку машин.

8.2. При распараллеливании счета координат кординаты каждой точки равномерно распределяются между машинами. Все машины работают независимо, время от времени информируя друг друга о точках, исключенных по своим координатам (пересылают друг другу выделители У). Эти пересылки, по-видимому, целесообразно делать после обработки группы координат.

В данном варианте распараллеливания характер исключения точек в процессе счета не вызывает неравномерностей в загрузке машин, что делает его предпочтительнее распараллеливания по точкам.

В заключение отметим, что процесс решения задач, подобных рассмотренной, естественным образом распараллеливается между машинами системы. Процесс решения заметно ускоряется,если применять в качестве решающих функций гиперплоскости, перпендикулярные осям координат. Это позволяет прекращать счет на первой же координате, для которой не выполняется условие, и эффективно вести одновременный счет для многих точек с помощью логических операций. Для этого исходная информация записывается не по словам, а по разрядам (вертикально). При вертикальной записи достигается компактность и единообразие, что позволяет лучше использовать оперативную память, упростить программирование и подготовку данных.

Авторы выражают глубокую признательность В.Н.Елкиной за полезные обсуждения и замечания.

Л И Т Е Р А Т У Р А

1. В.Н. ЕЛКИНА, Н.Г. ЗАГОРУЙКО. Об алфавите объектов распознавания. - Вычислительные системы, Новосибирск, Изд-во "Наука" СО, 1966, вып.22, стр.59-76.

2. В.Н. ЕЛКИНА, Н.Г. ЗАГОРУЙКО. Количественные критерии качества таксономии и их использование в процессе принятия решений. - Вычислительные системы, Новосибирск, Изд-во "Наука" СО,1969, вып.36, стр.29-46.

3. Э.В. ЕВРЕИНОВ, Г.П. ЛОПАТО. Универсальная вычислительная система "Минск-222". - Вычислительные системы, Новосибирск, Изд-во "Наука" СО, 1966, вып.23, стр.13-20.

4. Н.Г. ЗАГОРУЙКО. Сравнение решающих функций по мощности и затратам. - Вычислительные системы, Новосибирск,Изд-во "Наука" СО, 1969, вып.37, стр.10-14.

5. Ю.Г. КОСАРЕВ. Распараллеливание по циклам. - Вычислительные системы, Новосибирск, Изд-во "Наука" СО, 1967, вып.24, стр.3-20.

Поступила в редакцию
27 июля 1970 г.