А. Н. Велигура, Д. С. Симоненкова

О ЗАДАЧЕ СОСТАВЛЕНИЯ МНОГОПРОЦЕССОРНОГО РАСПИСАНИЯ, МИНИМИЗИРУЮЩЕГО ВРЕМЯ ВЫПОЛНЕНИЯ ШИФРАЛГОРИТМА AES

1. Общая постановка задачи составления многопроцессорного расписания

Рассматривается задача выполнения операций $v_1,\ldots,v_n$ на процессорах $p_1,\ldots,p_m$ за возможно меньшее время при следующих условиях (ограничениях).

Каждая операция принадлежит множеству возможных операций $A=\{a_1,\ldots,a_k\}$, т. е. $v_i\in A, i=1,\ldots,n$. Обозначим через n_s число операций a_S в списке $v_1,\ldots,v_n$:

$$n_S=\left|\left\{i\middle|v_i=a_S\right\}\right|, s=1,\ldots k. \tag{1.1}$$

Каждый процессор может быть предварительно запрограммирован (настроен) на выполнение одной из операций из множества A (возможно, не любой). Время выполнения операции v_i на процессоре p_j считается известным для всех пар $i=1,\ldots,n$, $j=1,\ldots,m$.

Некоторые операции используют результаты других операций, т. е. должны выполняться по завершении последних и передачи данных от них. Время, требуемое для передачи данных от одного процессора к другому, считается известным (для любой пары процессоров).

Задача состоит в назначении процессоров для выполнения операций $v_1,\ldots,v_n$ таким образом, чтобы минимизировать время выполнения всех операций с учетом возможности процессоров выполнять те или иные операции и необходимости передавать результаты одних операций для выполнения других.

1.2. Формализация постановки задачи

Одним из возможных подходов к решению задачи составления является формулировка ее в виде задачи математического программирования, т. е. задачи поиска минимума некоторой функции при условии, что ее аргумент должен удовлетворять определенным уравнениям или неравенствам [1]. Для формализации постановки задачи введем следующие обозначения.

Пусть вычисления начинаются в нулевой момент времени. Обозначим через t_i время начала выполнения i-й операции, $i=1,\ldots,n$.

Каждый процессор требуется настроить на выполнение какой-то одной операции из множества A. Введем переменные γ_{ij}, $i=1,\ldots,k$, $j=1,\ldots,m$; положим $\gamma_{ij}=1$, если процессор p_j запрограммирован на выполнение операции вида a_i, в противном случае $\gamma_{ij}=0$. Так как каждый процессор может быть настроен на выполнение только одной операции, то

$$\sum_{i=1}^{k}\gamma_{ij}=1,\ j=1,\ldots,m. \tag{1.2}$$

Если операция a_i не может выполняться на процессоре p_j (например, по техническим причинам), то в условия задачи добавляется уравнение $\gamma_{ij}=0$.

Для каждой операции из множества $v_1,\ldots,v_n$ необходимо выбрать процессор, на котором она будет выполняться. Введем переменные δ_{ij}, $i=1,\ldots,n$, $j=1,\ldots,m$; положим $\delta_{ij}=1$, если операция v_i будет выполняться на процессоре p_j, в противном случае положим $\delta_{ij}=0$. Так как каждая операция будет выполняться только на одном процессоре, то

$$\sum_{j=1}^{m}\delta_{ij}=1,\ i=1,\ldots,n. \tag{1.3}$$

Операция v_i может выполняться на процессоре p_j только в том случае, если он соответствующим образом запрограммирован, т. е. если $v_i=a_r$, то при $\gamma_{rj}=0$ (j-й процессор не запрограммирован на выполнение операции a_r) обязательно должно быть $\delta_{ij}=0$ (нельзя назначать выполнение операции v_i на процессоре p_j). При этом при $\gamma_{rj}=1$ на δ_{ij} ограничений

не накладывается, так как в этом случае операция v_i может как выполняться, так и не выполняться на процессоре p_j. Отсюда следует, что для каждого $r = 1,\ldots,k$ и для всех i, таких, что $v_i = a_r$, должны выполняться неравенства

$$\delta_{ij} - \gamma_{rj} \le 0;\ r = 1,\ldots,k,\ i : v_i = a_r,\ j = 1,\ldots,m. \tag{1.4}$$

Обозначим через τ_{ij} время выполнения операции v_i на процессоре p_j, $i = 1,\ldots,n$, $j = 1,\ldots,m$. Эти величины считаются известными. Исходя из этого время выполнения операции v_i в нашем расписании есть

$$\sum_{j=1}^{m} \tau_{ij}\delta_{ij},\ i = 1,\ldots,n.$$

Тогда, учитывая, что начало выполнения операции v_i есть t_i, получим, что она будет завершена в момент

$$t_i + \sum_{j=1}^{m} \tau_{ij}\delta_{ij},\ i = 1,\ldots,n. \tag{1.5}$$

Если результат операции v_i используется при выполнении операции v_r, то время начала выполнения операции v_r должно быть больше либо равно времени окончания выполнения операции v_i плюс время d_{js} на передачу данных от процессора p_j, если на нем выполнялась операция v_i, к процессору p_s, если на нем выполнялась операция v_r:

$$t_r - t_i \ge \tau_{ij} + d_{js} - M(2 - \delta_{ij} - \delta_{rs}),\ j,s = 1,\ldots,m,(i,r) \in B, \tag{1.6}$$

где B — множество пар i,r, таких, что результат операции v_i используется операцией v_r.

Если v_i и v_r — одинаковые операции и результат операции v_i используется при выполнении операции v_r, то при $j = s$ эти операции предлагается выполнять на одном и том же процессоре. Если на процессоре p_s это по каким-то причинам невозможно, то из системы неравенств (1.6) следует исключить неравенства, полученные при $j = s$ для каждого такого s.

Здесь M — достаточно большое положительное число, подбираемое из тех соображений, чтобы при $\delta_{ij} + \delta_{rs} < 2$ неравенство (1.6) превращалось в верное при любых возможных значениях переменных, т. е. фактически не создавало ограничений.

Если v_i и v_j — одинаковые операции, то нужно потребовать, чтобы они не были назначены для выполнения на одном процессоре одновременно. Для всех таких пар операций введем специальные переменные λ_{ij}. Переменная λ_{ij} — индикатор того, что операция v_i начнет выполняться позже операции v_j. Тогда для таких i, j

$$t_j - t_i \ge \tau_{ik} - M(2 + \lambda_{ij} - \delta_{ik} - \delta_{jk}),\ k = 1,\ldots,m, \tag{1.7}$$

$$t_i - t_j \ge \tau_{ik} - M(3 - \lambda_{ij} - \delta_{ik} - \delta_{jk}),\ k = 1,\ldots,m. \tag{1.8}$$

1.3. Задача составления расписания

С учетом введенных обозначений задача составления расписания, минимизирующего время выполнения операций $v_1,\ldots,v_n$ на процессорах $p_1,\ldots,p_m$, формулируется в следующем виде:

найти min t при условиях:

$$t_i + \sum_{j=1}^{m} \tau_{ij}\delta_{ij} - t \le 0,\ i = 1,\ldots,n \tag{1.9}$$

$$t_i - t_r + M\delta_{ij} + M\delta_{rs} \le 2M - \tau_{ij} - d_{js}\quad j,s = 1,\ldots,m\ (i,r) \in B \tag{1.10}$$

$$\begin{gathered} t_i - t_j + M\delta_{ik} + M\delta_{jk} - M\lambda_{ij} \le 2M - \tau_{ik} \\ k = 1,\ldots,m,\ s = 1,\ldots,k,\ s = 1,\ldots,k,\ i,j : \nu_i = \nu_j = a_S \end{gathered} \tag{1.11}$$

$$t_j - t_i + M\delta_{ik} + M\delta_{jk} + M\lambda_{ij} \le 3M - \tau_{ik}\ k = 1,\ldots,m\ , s = 1,\ldots,k\ , i, j : \nu_i = \nu_j = a_S \quad (1.12)$$

$$\delta_{ij} - \gamma_{rj} \le 0\ ,\ r = 1,\ldots,k\ ,\ i : \nu_i = a_r\ ,. \quad (1.13)$$

$$\sum_{j=1}^{m} \delta_{ij} = 1\ , i = 1,\ldots,n\ \ j = 1,\ldots,m \quad (1.14)$$

$$\sum_{i=1}^{k} \gamma_{ij} = 1\ ,\ j = 1,\ldots,m \quad (1.15)$$

где все переменные — целые неотрицательные, величины

τ_{ij} — время выполнения операции v_i на процессоре p_j,

d_{is} — время передачи данных от процессора p_j к процессору p_s — заданы,

M — достаточно большое положительное число, задается по результатам предварительной оценки возможных диапазонов изменения переменных.

Общее число переменных равно

$$1 + n + km + nm + \sum_{s=1}^{k} \binom{n_s}{2}.$$

Общее количество ограничений (уравнений и неравенств) равно

$$n + m^2|B| + m\sum_{s=1}^{k} n_s^2 + n + m = 2n + m\left(m|B| + \sum_{s=1}^{k} n_s^2 + n + 1\right).$$

Общее число ненулевых элементов в матрице ограничений равно

$$n(m+2) + 4m^2|B| + 10m\sum_{s=1}^{k}\binom{n_s}{2} + 3mn + km = 2n + 4m^2|B| + 5m\sum_{s=1}^{k} n_s^2 - mn + km =$$
$$= 2n + m\left(4m|B| + 5\sum_{s=1}^{k} n_s^2 - n + m\right)$$

Сравнивая число ненулевых коэффициентов матрицы с общим числом коэффициентов, мы видим, что матрица ограничений при больших n является разреженной. Разреженность матриц позволяет использовать при работе с ними специальные приемы понижения трудоемкости основных матричных операций [2] по сравнению с общим случаем, когда разреженность отсутствует.

2. Пример построения ЗЦЛП для AES-128

AES — итеративный обратимый блочный шифр с варьируемыми размерами ключа, блоков информации и числом однородных раундов шифрования.

Входные, промежуточные и выходные блоки данных (состояния) удобно представить в виде квадратных массивов байтов (матриц) S = (a_{ij}); i = 0,…,3; j = 0,…,3. Ключ раунда также можно представить в виде прямоугольного массива байтов размером 4 на 4 [3].

Каждый раунд состоит в последовательном выполнении следующих преобразований независимо для каждой из 4 групп по 4 байта:

1. замена каждого байта текущего состояния независимо от других по таблице замены S — 4 операции замены;

2. умножение строки из 4 байтов на заданную матрицу над полем GF(256) (в каждом столбце матрицы два элемента равны единице) — 8 операций умножения и 12 операций сложения в поле (в последнем раунде это преобразование отсутствует);

3. побитное сложение байтов текущего состояния с байтами раундового ключа по модулю 2 — 4 операции.

Число видов байтовых операций равно четырем (замена, умножение в поле, сложение в поле, побитное сложение), общее число операций для 4 байтов составляет 4 замены, 8 умножений и 12 сложений в поле, 4 побитных сложения — всего 28 операций. Пронумеровав эти операции

в вышеприведенном порядке от 1 до 28, получим следующее множество пар B, множество пар i,r, таких, что результат операции i используется операцией j:

B={{1,5},{1,12},{2,6},{2,7},{3,8},{3,9},{4,10},{4,11},{5,13},{6,13},{13,14},{3,14},
{14,15},{1,16},{7,16},{16,17},{8,17},{17,18},{4,18},{1,19},{2,19},{19,20},{9,20},
{20,21},{10,21},{12,22},{2,22},{22,23},{3,23},{23,24},{11,24},{15,25},{18,26},
{21,27},{24,28}}.

После задания числа процессоров, времени выполнения операций на процессорах и передачи данных будут получены все данные для задачи составления расписания (1.9)–(1.15), минимизирующего время выполнения алгоритма AES.

Заключение

Задача целочисленного линейного программирования в общем случае NP-полна [4], поэтому практическое решение полученной задачи построения расписания, минимизирующего время выполнения шифралгоритма AES, при данных ее размерах если и возможно, то лишь при наличии у нее каких-то специальных свойств. Выявление таких свойств может быть предметом дополнительных исследований.

СПИСОК ЛИТЕРАТУРЫ:

1. Кун С. Матричные процессоры на СБИС. М.: Мир, 1991. – 627 с.
2. Announcing the ADVANCED ENCRYPTION STANDARD (AES) / Federal Information Processing Standards Publication 197, November 26, 2001. – 51 p.
3. Писсанецки С. Технология разреженных матриц. М.: Мир, 1988. – 410 с.
4. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. – 416 с.