

Оценка локальности параллельных алгоритмов, реализуемых на графических процессорах

Полещук Максим Александрович¹

Лиходед Николай Александрович, д. ф.-м. н., профессор²

г.Минск, БГУ, ФПМИ^{1,2}

maxipole@gmail.com¹

likhoded@bsu.by²

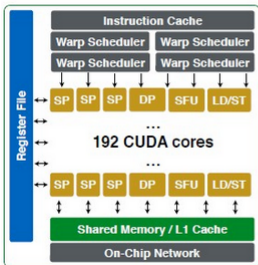
29 марта 2016 г.

Постановка задачи

- Получить блоки операций и потоки операций параллельного алгоритма, приводящих к меньшему числу обращений к глобальной памяти и к эффективному использованию параллельными потоками вычислений кэшей и разделяемой памяти графического процессора.
- Минимизировать число промахов кэша за счет использования временной и пространственной локальности данных с учетом размера и длины строк кэша.

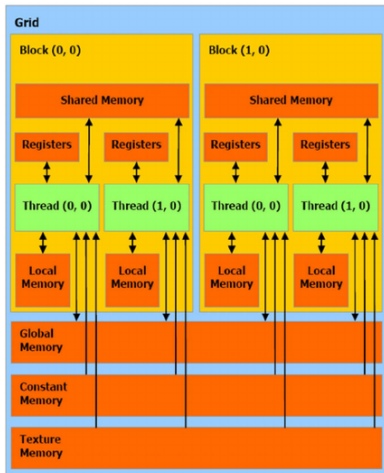
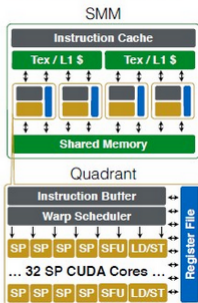
Параллельные вычисления на графических процессорах

Kepler SM (SMX)



© THENEXTPLATFORM.COM

Maxwell SMM



Пример 1. Основная часть алгоритма Флойда-Уоршелла поиска кратчайших путей между всеми парами вершин графа

```
for  $k = 1, n$ 
  for  $i = 1, n$ 
    for  $j = 1, n$ 
 $S_1: a(i, j) = \min(a(i, j), a(i, k) + a(k, j))$ 
    end
  end
end
```

3D блоки блочного алгоритма, для которых $r_1 = r_2 = r_3 = r > 1$:

```
for  $k = 1 + k^{block} * r, \min((k^{block} + 1) * r, n)$ 
  for  $i = 1 + i^{block} * r, \min((i^{block} + 1) * r, n)$ 
    for  $j = 1 + j^{block} * r, \min((j^{block} + 1) * r, n)$ 
 $S_1: a(i, j) = \min(a(i, j), a(i, k) + a(k, j))$ 
    end
  end
end
```

Пример 2. Основная часть алгоритма прямого хода метода Гаусса (без выбора ведущего элемента)

```
for  $k = 1, n - 1$   
  for  $i = k + 1, n$   
    for  $j = k + 1, n + 1$   
       $S_1: a(i, j) = a(i, j) - a(i, k)a(k, j)/a(k, k)$   
    end  
  end  
end
```

3D блоки блочного алгоритма:

```
for  $k = 1 + k^{block} * r_1, \min((k^{block} + 1) * r_1, n - 1)$   
  for  $i = \max(2 + i^{block} * r_2, k + 1), \min(1 + (i^{block} + 1) * r_2, n)$   
    for  $j = \max(2 + j^{block} * r_3, k + 1), \min(1 + (j^{block} + 1) * r_3, n + 1)$   
       $S_1: a(i, j) = a(i, j) - a(i, k)a(k, j)/a(k, k)$   
    end  
  end  
end
```

Пусть в гнезде циклов имеется Θ наборов выполняемых операторов.

Зависимости (информационные связи) $S_\alpha(I) \rightarrow S_\beta(J)$ между операциями будем характеризовать векторами $d^{\alpha,\beta} = J - I$.

Пример 1: $d^{a(i,j)a(i,j)} = (1, 0, 0)$, $d^{a(i,j)a(i,k)} = (1, 0, j - k)$,
 $d^{a(i,j)a(k,j)} = (1, i - k, 0)$.

$d_\zeta^{\alpha,\beta,\max}$ – максимальные по модулю значения компонент векторов $d^{\alpha,\beta}$.

Число частей Q_ζ^ϑ , на которые при формировании блоков разбивается область значений параметра j_ζ цикла, окружающего ϑ -й набор операторов:

$$Q_\zeta^\vartheta = \lceil (M_\zeta^\vartheta - m_\zeta^\vartheta + 1) / r_\zeta^\vartheta \rceil.$$

Предельные значения параметров циклов:

$$M^\vartheta = \max_\zeta (M_\zeta^\vartheta - m_\zeta^\vartheta) + 1.$$

Определения (продолжение)

Функция доступа для вхождения массива в правую часть оператора:

$$\bar{F}_{\ell,\beta,q}(J) = F_{\ell,\beta,q}J + G_{\ell,\beta,q}N + f^{\ell,\beta,q}.$$

	$a(i, j)$	$a(i, k)$	$a(k, j)$
$F_{\ell,\beta,q}$	$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
$G_{\ell,\beta,q}$	0		
$f^{\ell,\beta,q}$	0		

«Фиксированное данное массива» – конкретное, неизмененное содержимое соответствующей ячейки памяти.

Величины $\rho_{\ell,\beta,q}^d, \rho_{\ell,\beta,q}^{d,\zeta}$. Пусть число фиксированных данных, используемых на вхождении (ℓ, β, q) , оценивается величиной $O(M^{\rho_{\ell,\beta,q}^d})$, а число фиксированных данных, используемых на вхождении (ℓ, β, q) при фиксированном значении цикла с параметром j_ζ , оценивается величиной $O(M^{\rho_{\ell,\beta,q}^{d,\zeta}-1})$.

Теорема 1

Пусть вхождение (ℓ, β, q) порождает истинную зависимость: определение некоторого данного происходит на вхождении $(\ell, \alpha, 1)$ в левой части оператора S_α , а использование – на вхождении (ℓ, β, q) в правой части оператора S_β , причем в окружении операторов S_α и S_β имеется цикл с параметром j_ζ . Тогда, при получении блоков вычислений для реализации алгоритма на графическом процессоре, разбиение итераций цикла j_ζ порождает коммуникационные операции чтения и записи, объем которых имеет следующие оценки:

- не требуется ни операций чтения, ни операций записи, если $d_\zeta^{\alpha, \beta, \max} = 0$;
- $O(Q_\zeta^\vartheta M^{\rho_{\ell, \beta, q}^d})$ операций чтения и операций записи, если $0 < d_\zeta^{\alpha, \beta, \max} < r_\zeta^\vartheta$;
- $O(Q_\zeta^\vartheta M^{\rho_{\ell, \beta, q}^d})$ операций чтения и $O(M^{\rho_{\ell, \beta, q}^d})$ операций записи, если $d_\zeta^{\alpha, \beta, \max} \geq r_\zeta^\vartheta, \rho_{\ell, \beta, q}^{d, \zeta} \neq \rho_{\ell, \beta, q}^d$;
- $O(M^{\rho_{\ell, \beta, q}^d})$ операций чтения и операций записи, если $d_\zeta^{\alpha, \beta, \max} \geq r_\zeta^\vartheta, \rho_{\ell, \beta, q}^{d, \zeta} = \rho_{\ell, \beta, q}^d$.

Теорема 1 (продолжение)

В случае, когда вхождение (ℓ, β, q) не порождает истинной зависимости (происходит обращение к входным данным) или цикл с параметром j_ζ имеется в окружении только оператора S_β , оценки следующие:

- $O(Q_\zeta^{\vartheta} M^{\rho_{\ell, \beta, q}^d})$ операций чтения, если $\rho_{\ell, \beta, q}^{d, \zeta} \neq \rho_{\ell, \beta, q}^d$;
- $O(M^{\rho_{\ell, \beta, q}^d})$ операций чтения, если $\rho_{\ell, \beta, q}^{d, \zeta} = \rho_{\ell, \beta, q}^d$.

Лемма

Пусть для вхождения (ℓ, β, q) массива a_ℓ в правую часть оператора S_β для всех $\xi \notin \Xi^{\ell, \beta, q}, \xi \leq \xi^{\ell, \beta, q}$, выполнены условия $r_{\xi, 2} = 1$ (условие накладывается при наличии указанных ξ) и $\nu_\ell \geq |\Xi^{\ell, \beta, q}|$. Тогда в каждом потоке вычислений на разных итерациях циклов с параметрами $j_\xi, 1 \leq \xi \leq \xi^{\ell, \beta, q}$, используются разные элементы массива a_ℓ .

Пример 1 (алгоритм Флойда-Уоршелла). В таблице указаны координаты, размер которых можно задать произвольным образом (в пределах размера блока), так как число кэш-промахов на этом вхождении будет одинаково по всем потокам вычислений:

Вхождение	Матрица	$\Xi^{\ell, \beta, q}$
$a(i, j)$	$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\{2, 3\}$
$a(i, k)$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$	$\{1, 2\}$
$a(k, j)$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\{1, 3\}$

Определения (продолжение)

$\xi^{\ell,\beta,q}$ — номер ненулевого столбца матрицы $F_{\ell,\beta,q}$, правее которого находятся только нулевые столбцы матрицы; если самый правый столбец $F_{\ell,\beta,q}$ — ненулевой, то по определению $\xi^{\ell,\beta,q} = n_\beta$.

Пример 1 (продолжение).

Вхождение	Матрица	$\xi^{\ell,\beta,q}$
$a(i, k)$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$	2
$a(k, j)$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	3

$\Xi^{\ell,\beta,q}$ — множество номеров линейно независимых столбцов матрицы $F_{\ell,\beta,q}$ (если таких множеств более одного, то зафиксировать любое);

L_ℓ — число элементов массива a_ℓ , помещающихся в строку кэша; предполагается, что строка массива помещается в кэш;

ν_ℓ — размерность массива a_ℓ .

Теорема 2 (о наименьшем совокупном числе кэш-промахов по всем потокам вычислений)

Пусть выполняются предположения леммы и не более чем один столбец $F_{\ell, \beta, q, \zeta}$ имеет вид $(0, \dots, 0, b_{\ell, \beta, q, \zeta})$, $0 < |b_{\ell, \beta, q, \zeta}| r_{\zeta, 2} < L_{\ell}$. Если $r_{\zeta, 2} = 1$, цикл с параметром j_{ζ} выполняется параллельно, циклы с параметрами j_{ξ} , $\xi \notin \Xi^{\ell, \beta, q}$, $F_{\ell, \beta, q} \neq 0$, выполняются последовательно (с синхронизацией потоков вычислений между итерациями в случае $r_{\xi} > 1$), то независимо от выбора $r_{\xi, 2}$, $\xi \in \Xi^{\ell, \beta, q}$, совокупное число кэш-промахов по всем потокам вычислений является наименьшим (условия накладываются при наличии указанных столбцов).

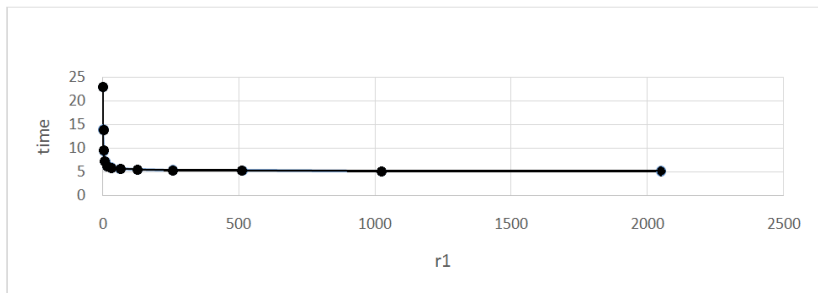
Пример 2 (алгоритм метода Гаусса).

Вхождение	Матрица	ζ
$a(i, j)$	$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	3
$a(i, k)$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$	—
$a(k, j)$	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	3

Пример 1 (продолжение). Объём коммуникационных операций

	$a(i,j)$	$a(i,k)$	$a(k,j)$
k	$d_1^{\max} = 1, O(Q_1 M^{\rho_{a(i,j)}^d - 1})$		
	$O(Q_1 n^2)$	$O(Q_1 n)$	$O(Q_1 n)$
i	$d_2^{\max} = 0$ 0	$d_2^{\max} = 0$ 0	$d_2^{\max} \geq r_2, \rho_{a(k,j)}^{d,2} \neq \rho_{a(k,j)}^d$ $O(Q_2 M^{\rho_{a(k,j)}^d}) = O(Q_2 n^2)$ <u>$O(M^{\rho_{a(k,j)}^d}) = O(n^2)$</u>
j	$d_3^{\max} = 0$ 0	$d_3^{\max} \geq r_3, \rho_{a(k,j)}^{d,3} \neq \rho_{a(k,j)}^d$ $O(Q_3 M^{\rho_{a(k,j)}^d}) = O(Q_3 n^2)$ <u>$O(M^{\rho_{a(k,j)}^d}) = O(n^2)$</u>	$d_3^{\max} = 0$ 0

Пример 2. Зависимость времени реализации блочного алгоритма Гаусса от r_1



r_1 – размер блока по координате k

GeForce GT 860M	
Глобальная память	4295 МБ
Разделяемая память	49 КБ
Количество потоков в варпе	32
Мультипроцессоров	5
Всего ядер	640

Направления дальнейших исследований

- разработка и программная реализация алгоритмов, позволяющих выбирать соотношения размеров блоков вычислений
- получение условий и соотношений, точно характеризующих объем коммуникационных операций
- исследования, направленные на минимизацию объема коммуникационных операций между параллельными вычислительными процессами, реализуемыми на суперкомпьютерах с распределенной памятью
- разработка и программная реализация параллельных алгоритмов для решения прикладных задач с использованием предлагаемого подхода

- Лиходед Н.А., Полещук М.А. Метод ранжирования параметров размера блоков вычислений параллельного алгоритма // Доклады НАН Беларуси. 2015. Т. 59, № 4. С. 25–33.
- Kandemir M., Ramanujam J., Irwin M., Narayanan V., Kadayif I., Parikh A. A compiler based approach for dynamically managing scratch-pad memories in embedded systems // IEEE Transactions on Computer-Aided Design. 2004. Vol. 23, No. 2. P. 243–260.
- Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. Санкт-Петербург: БХВ-Петербург, 2002. 608 с.
- Katz G.J., Kider J. All-pairs shortest-paths for large graphs on the GPU // Proceedings of the 23rd ACM SIGGRAPH/EUROGRAPHICS symposium on Graphics hardware. Sarajevo, Bosnia and Herzegovina: Eurographics Association. 2008. P. 47–55.
- Lund B.D., Smith J.W. A multi-stage cuda kernel for floyd-warshall // CoRR abs/1001.4108. 2010.

Спасибо за внимание