Применение параллельных вычислений в задаче компьютерной томографии

А.Е. Ковтанюк[1,2], А.А. Хандорин[1]

[1]*Дальневосточный Федеральный Университет, Суханова 8, Владивосток, Россия*
[2]*Институт прикладной математики ДВО РАН, Радио 7, Владивосток Россия*

kovtanyuk.ae@dvfu.ru, alex.khandorin@gmail.com

Аннотация. *В работе рассмотрена задача компьютерной томографии, заключающаяся в восстановлении структуры трехмерного объекта по данным радиационного просвечивания. Решение задачи представляется в виде набора двумерных изображений. Реконструкция объекта в отдельном сечении осуществляется на основе алгоритма свёртки и обратной проекции. Параллелизация вычислительного процесса осуществляется по набору сечений. Программная реализация алгоритма осуществляется на основе технологии MPI.*

Ключевые слова

Алгоритм свертки и обратной проекции, параллельные вычисления, компьютерная томография, MPI.

1 Введение

Восстановление структуры трехмерного объекта по данным радиационного просвечивания является важной задачей компьютерной томографии. Обычно в результате сканирования измеряется выходящее излучение на наборе прямых, лежащих в отдельных сечениях. При этом восстановление изображения сводится к решению задачи обращения преобразования Радона. Алгоритм свертки и обратной проекции выделяется среди наиболее популярных алгоритмов, решающих эту задачу, своим относительным быстродействием. К примеру, реконструкция изображения размером 400 x 400 пикселей, полученная по данным просвечиваний по 10201 прямым, требует около 2 секунд машинного времени при использовании процессора Intel(R) Xeon(R) CPU E5345 @ 2.33GHz.

Однако, восстановление трехмерного изображения объекта по набору сечений в более высоком разрешении и с большим количеством просвечиваний займёт существенное время. Так при реконструкции 256 изображений размером 1024 x 1024 пикселей каждое, при использовании просвечиваний по 40401 прямым, время вычислений на упомянутом выше процессоре увеличится более чем в 1 000 раз. Таким образом, использование параллельных вычислений при реконструкции структуры трехмерного объекта представляется весьма уместным.

Эффективная параллелизация алгоритмов, решающих задачу компьютерной томографии, может осуществляться для алгебраических методов обращения преобразования Радона [1], а также Фурье-алгоритма [2]. В настоящей работе авторами обсуждаются перспективы применения параллельных вычислений при реализации алгоритма свертки и обратной проекции.

2 Теоретические аспекты

Для описания алгоритма свертки и обратной проекции введем ряд понятий.

Пространство Шварца $\mathcal{G}(R^2)$, или $\mathcal{G}$, представляет собой линейное пространство определенных на R^2 бесконечно дифференцируемых функций, для которых норма

$$|f|_{k,l} = \sup_{x\in R^2} |x^k D^l f(x)|$$

конечна для всех индексов $k, l \in Z_+^2$.

Двумерное преобразование Радона $\mathbf{R}$ отображает функцию, определенную в R^2, во множество ее линейных интегралов по прямым в R^2. Пусть $\theta \in S^1$, где S^1 - множество единичных векторов, и $s \in R^1$, тогда

$$\mathbf{R}f(\theta,s) = \int\limits_{x\cdot\theta=s} f(x)\,dx$$

представляет собой интеграл функции f, принадлежащей пространству Шварца $\mathcal{G}(R^2)$, по прямой, перпендикулярной вектору θ и расположенной на расстоянии s от начала координат. $\mathbf{R}f$ – четная функция, определенная на единичном цилиндре

$Z = S^1 \times R^1$.

Оператор обратного проектирования вводится следующим образом:

$$\mathbf{R}^{\#}g(x) = \int\limits_{S^1} g(\theta, x\cdot\theta)\,d\theta.$$

Операторы $\mathbf{R}$ и $\mathbf{R}^{\#}$ образуют двойственную пару в смысле интегральной геометрии: оператор $\mathbf{R}$ задает интегрирование по всем точкам плоскости, а оператор задает интегрирование по всем плоскостям, проходящим через данную точку.

Если $h, g \in \mathcal{G}(Z)$, тогда операция свертки для функции g задаётся формулой:

$$[h * g](x) = \int\limits_{R^1} h(\theta, s-t)g(\theta,t)\,dt.$$

Операция свертки для функции, определенной на Z, всегда выполняется по второй переменной.

Метод свертки и обратной проекции обращения преобразования Радона основывается на следующей формуле [3]

$[W_b * f] = \mathbf{R}^{\#}[w_b * \mathbf{R}f]$, $W_b = \mathbf{R}^{\#}w_b$.

Идея алгоритма заключается в таком выборе w_b, чтобы функция W_b аппроксимировала δ - функцию. Точнее, функция W_b должна представлять собой фильтр низких частот с предельной частотой b:

$$\hat{W_b}(\xi) = \frac{1}{2\pi}\hat{\Phi}\left(\frac{|\xi|}{b}\right),$$

где $0 \le \hat{\Phi} \le 1$ и $\hat{\Phi}(\sigma) = 0$ при $\sigma \ge 1$. Таким образом, $W_b \to \delta$ при $b \to 0$.

Для вычисления функции f нужно выполнить одномерную операцию свертки, или фильтрации, $w_b * \mathbf{R}f$ для каждого направления из S^1, а затем применить оператор обратного проектирования $\mathbf{R}^{\#}$.

Пусть g - массив данных, полученных при сканировании исследуемого объекта под различными углами. Функция $g = \mathbf{R}f$ определена и задана в точках (θ_j, s_l), $j = 1,\dots,p$, $l = -q,\dots,q$, где $\theta_j \in S^1$ и $s_l = hl$, $h = 1/q$.

Свертка $[w_b * \mathbf{R}f]$ заменяется дискретной сверткой

$$[w_b * g(\theta_j, s)]_h = h\sum_{l=-q}^{q} w_b(s - s_l)\,g(\theta_j, s_l)\ .$$

Для вычисления обратной проекции используется квадратурная формула интегрирования по S^1 с узлами $\theta_1,\dots,\theta_p$ и положительными весами α_{pj} с тем требованием, чтобы эта формула была точна для функций $v \in H'_{2m}$ при некотором m, где H'_{2m} –множество четных сферических гармоник степени не выше $2m$, то есть

$$\int\limits_{S^1} v(\theta)\,d\theta = \sum_{j=1}^{p} \alpha_{pj}\,v(\theta_j)$$

для $v \in H'_{2m}$. Обратное проектирование заменяется дискретным обратным проектированием:

$$\mathbf{R}_p^{\#}v(x) = \sum_{j=1}^{p} \alpha_{pj}\,v(\theta_j, x\cdot\theta_j).$$

Теперь алгоритм принимает вид

$$f = \mathbf{R}_p^{\#}[w_b * g]_h.$$

В данном случае известны значения функции $g = \mathbf{R}f(\theta, s)$ в точках (θ_j, s_l), $j = 1, \dots, p$, $l = -q, \dots, q$, $\theta_j = (\cos\varphi_j, \sin\varphi_j)$, $\varphi_j = \pi(j-1)/p$, $s_l = hl$, $h = 1/q$.

Если воспользоваться на S^1 квадратурной формулой трапеций с p узлами, которая точна на H'_{2p-2}, и применить линейную интерполяцию, то алгоритм примет вид.

Шаг 1. Для $j = 1, \dots, p$ вычисляются свертки

$$v_{j,k} = h\sum_{l=-q}^{q} w_b\,(s_k - s_l)\,g(\theta_j, s_l), \quad l = -q, \dots, q,$$

где для определения функции w_b используется представление [4]

$$w_b(s) = \frac{b^2}{2\pi^3}u(bs),$$

где

$$u(s) = \frac{\frac{\pi}{2} - s\sin s}{\left(\frac{\pi}{2}\right)^2 - s^2}.$$

Если $b = \pi/h$ и $s = s_l$, формула упрощается:

$$w_b(s_l) = \frac{1}{h^2\pi^2}\frac{1}{1-4l^2}.$$

Шаг 2. В каждой точке восстановления x вычисляется дискретная обратная проекция

$$f(x) = \frac{2\pi}{p}\sum_{j=1}^{p}\left((1-u)v_{j,k} + uv_{j,k+1}\right),$$

где k и u для каждой пары x, j задаются формулами

$$s = \theta_j \cdot x, \quad k \le \frac{s}{h} < k+1, \quad u = \frac{s}{h} - k.$$

Изложенный алгоритм будет надежно восстанавливать функции с эффективной шириной спектра b при достаточно большом b, удовлетворяющем условиям $b < p$ и $h \le \pi/b$.

Вычисление сверток требует $O(pq^2)$ операций. Обратное проектирование требует $O(p)$ для каждой точки x, то есть в общей сложности $O(pq^2)$ операций, если f вычисляется на сетке $(2q+1)(2q+1)$. При оптимальном соотношении $p = \pi q$ [3] для реализации алгоритма требуется $O(q^3)$ операций.

3 Параллельная реализация алгоритма

Для параллельной реализации алгоритма свёртки и обратной проекции авторами было использовано следующее решение: были выделены расчётные процессы и управляющий процесс, не использующий большие вычислительные мощности.

Алгоритм работы управляющего процесса:

Шаг 1. Обнуляются счётчики количества срезов и расчётных процессов.

Шаг 2. Если значение счётчика количества срезов меньше либо равно общего количества срезов, то оно инкрементируется.

Шаг 3. Ожидается запрос от одного из расчётных процессов. В ответ ему отправляется значение счётчика количества срезов.

Шаг 4. Если значение счётчика количества срезов больше общего количества срезов, то инкрементируется значение счётчика расчётных процессов.

Шаг 5. Если значение счётчика расчётных процессов равно общему количеству расчётных процессов, то выводится общее время работы и расчётных процессов, иначе осуществляется переход к Шагу 2.

Алгоритм работы расчётного процесса:

Шаг 1. Отправляется запрос управляющему процессу, в ответе приходит значение счётчика количества срезов.

Шаг 2. Если значение счётчика количества срезов больше общего количества срезов, то работа процесса завершается, иначе осуществляется переход к Шагу 3.

Шаг 3. С помощью алгоритма свёртки и обратной проекции восстанавливаются данные среза с порядковым номером равным значению счётчика количества срезов. Осуществляется переход к Шагу 1.

Данный алгоритм позволяет распределить нагрузку между процессами в зависимости от их быстродействия.

В качестве исходных данных использовались зашумленные результаты сканирования 256 сечений фантома Кормака [5], состоящих из 101 x 101 измерений каждое, находящихся в 256 файлах соответственно. Вычисления были осуществлены на кластере Дальневосточного федерального университета (CentOS 6, OpenMPI) при использовании 32-х двухъядерных процессоров 2 x Intel(R) Xeon(R) CPU E5345 @ 2.33GHz. Ниже приведена таблица 1 результатов работы программы, в которой, помимо экспериментальных данных, приведён расчёт доли параллельных инструкций в коде программы, основанный на хорошо известном законе Амдала:

$$S_p = \frac{1}{1 - \alpha + \frac{\alpha}{p}} ,$$

где S_p – ускорение, получаемое при использовании p процессов, а α – доля параллельных инструкций.

Таблица 1. Результаты работы программы

Количество процессов	*Итоговое время (сек.)*	*Экспериментальное ускорение*	*Доля параллельных инструкций*
1	482,135	1	1
2	241,738	1,99445	0,9972
4	121,005	3,98442	0,9987
8	62,1094	7,76267	0,9956
16	31,4126	15,3486	0,9972
32	17,2215	27,9961	0,9954
64	9,94997	48,4559	0,9949

4 Заключение

Основываясь на полученных экспериментальных данных и законе Амдала, можно считать, что данная реализация алгоритма свертки и обратной проекции достаточно успешно масштабируется. Это позволяет существенно уменьшить время обработки результатов сканирования, что является в таких областях, как медицинская томография, одним из важнейших требований при исследовании объектов.

Реализация данного алгоритма для параллельной схемы сканирования является первым этапом реализации алгоритмов реконструкции изображений с использованием параллельных вычислений. В ближайшее время планируется проделать аналогичную работу, используя библиотеки для параллельных вычислений на графических процессорах, а также реализовать другие алгоритмы восстановления.

5 Благодарности

Данная работа выполнена при поддержке Научного фонда Дальневосточного федерального университета, а также НОЦ «СКТ – Дальний Восток».

Литература

[1] С.В. Баранник, А.Л. Головинский, А.В. Демин, А.Л. Маленко, Ю.А. Петровский: О возможностях параллельной обработки данных в ГРИД-системе хранения медицинских изображений для повышения эффективности диагностики. *Second International Conference "Cluster Computing" CC 2013 (Ukraine, Lvov, June 3-5, 2013)*.

[2] И.С. Грузман, В.С. Киричук, В.П. Косых, Г.И. Перетягин, А.А. Спектор: Цифровая обработка изображений в информационных системах: Учебное пособие. *Новосибирск: Изд-во НГТУ*: 352 с, 2002.

[3] Ф. Наттерер: Математические аспекты компьютерной томографии. *М.: Мир*: 288 с, 1990.

[4] L.A. Shepp, B.F. Logan: The Fourier reconstruction of a head section. *IEEE Trans. on Nuclear Science 21*:21-43, 1974.

[5] A.M. Cormack. *J. Applied Physics 35*: 2908, 1964.