

Синхронный алгоритм параллельной реконструкции трехмерных изображений на основе системы обмена сообщениями MPI*

С. А. Золотарев¹, В. Л. Венгринович¹, С. И. Смагин^{2,†}

¹Институт прикладной физики НАНБ, Минск, Беларусь

²Вычислительный центр ДВО РАН, Хабаровск, Россия

[†]Контактный e-mail: smagin@ccfebras.ru

Одни из наиболее важных вычислительных задач — разработка и адаптация итерационных методов для решения сверхбольших разреженных систем алгебраических уравнений. К таким вычислительным задачам приводит задача итерационной параллельной реконструкции трехмерных изображений промышленных изделий. Важно, что итерационные методы решения вычислительных задач большой размерности реализуются на параллельных структурах намного эффективнее, чем прямые методы их решения. В этой работе описан синхронный параллельный алгоритм, основанный на использовании системы MPI для решения задачи реконструкции трехмерных изображений промышленных изделий.

Ключевые слова: итерационные методы, реконструкция изображений, томография, параллельные алгоритмы, MPI, воксельный параллелизм.

Библиографическая ссылка: Золотарев С.А., Венгринович В.Л., Смагин С.И. Синхронный алгоритм параллельной реконструкции трехмерных изображений на основе системы обмена сообщениями MPI // Вычислительные технологии. 2019. Т. 24, № 6. С. 30–39. DOI: 10.25743/ICT.2019.24.6.005.

Введение

Быстрое развитие вычислительной техники обеспечило появление на рынке большого числа различных суперкомпьютеров, обладающих не доступными ранее возможностями. Они успешно применяются для решения разнообразных вычислительных задач в науке и промышленности [1]. Обширный обзор параллельных алгоритмов можно найти в известных работах В.Н. Фаддеевой и Д.К. Фаддеева [2–4], В.В. Воеводина [5–7], Дж. Ортеги [8], Д.П. Бертсекаса [9].

Наиболее широкое распространение получили системы параллельного программирования на основе интерфейса MPI (Message Passing Interface). Идея MPI [10] исходно проста и очевидна. Она предполагает представление параллельной программы в виде множества параллельно исполняющихся процессов (программы процессов, как правило, разработаны на обычном последовательном языке программирования C++ или Фортран), взаимодействующих в ходе выполнения программы для передачи данных с помощью коммуникационных процедур.

*Title translation and abstract in English can be found on page 39.

© ИВТ СО РАН, 2019.

Разработка, отладка и сопровождение параллельных программ томографической реконструкции являются достаточно сложным делом, но все это окупается повышением быстродействия на несколько порядков и возможностью хранения проекционной матрицы в оперативной памяти процессорных узлов суперкомпьютера. Поэтому довольно много усилий исследователей сконцентрировано на решении трехмерных томографических задач на массивно параллельных вычислительных системах с гибридной архитектурой (например, OpenPOWER).

Одной из наиболее важных задач является разработка и адаптация итерационных методов для решения сверхбольших разреженных систем алгебраических уравнений, к которым и приводит трехмерная томографическая задача. Важно, что итерационные методы решения вычислительных задач большой размерности реализуются на параллельных структурах намного эффективнее, чем прямые методы их решения. Большинство алгоритмов, основанных на прямых методах решения, обладают существенной наследственной последовательной структурой и требуют значительного числа взаимодействий процессоров, которые не могут быть выполнены в параллельном режиме. Итерационные же методы требуют значительно меньшего числа взаимодействий подобного типа и сравнительно легко отображаются на параллельные вычислительные структуры. Не менее важно и то, что в большинстве случаев параллельные реализации классических итерационных методов более эффективны в отношении скорости вычислений.

Параллельное выполнение алгоритма заключается в том, что вычислительный процесс распределяется некоторым образом между группой процессоров. В зависимости от способа взаимодействия локальных процессоров различают два типа выполнения параллельных итерационных алгоритмов: синхронные и асинхронные. В первом случае предполагается, что процессоры заканчивают вычисления и обмениваются всеми необходимыми полученными результатами до начала новой итерации. Основным недостатком синхронных параллельных алгоритмов является то, что они требуют синхронизации итераций. Это весьма сложная задача, особенно при увеличении числа процессоров. Кроме того, скорость расчета ограничивается скоростью работы самого медленного процессора, а более быстрые процессоры большую часть времени проводят в стадии ожидания. Но тем не менее реализация такого рода параллельных алгоритмов может быть эффективно осуществлена с использованием стандарта MPI.

Основные формы параллелизма в алгоритмах компьютерной томографии: воксельный, проекционный и лучевой. Схематично выполнение параллельного процесса можно описать так: $PAR \{ \text{последовательность вычислительных операций} \}$. Воксельный параллелизм использует регулярность структуры восстанавливаемых данных в виде двумерного или трехмерного массива, заданного в соответствующих центральных точках вокселей трехмерной области реконструкции. Пусть $J = 1, 2, \dots, N$ — множество всех вокселей. Если алгоритм томографической реконструкции использует идентичные операции для каждого вокселя, то вычислительный процесс может быть распараллелен следующим образом: $PAR J = 1 FOR N \{ \text{вычислительный процесс для } j\text{-го вокселя} \}$.

1. Синхронный параллельный алгоритм реконструкции изображений

В лаборатории вычислительной диагностики ИПФ НАНБ разработаны синхронные параллельные вычислительные алгоритмы и программные коды на языке программиро-

вания C++ и системы MPI для трехмерной томографической реконструкции в коническом пучке. Отладка программного кода и численные расчеты осуществлены на гибридном вычислительном кластере на базе архитектуры OpenPOWER в системе MPI. Для осуществления параллельной трехмерной томографической реконструкции использована воксельная форма параллелизма.

Алгоритм итерационной параллельной реконструкции изображений при круговом движении источника излучения вокруг оси OY

Вход. Число итераций n_iter , Коэффициент релаксации λ , Число двумерных проекций — n_prj , Проекции $P^{(l)}$, $l = 1, \dots, n_prj$ размерностями по осям OX и OY — $size_x$ и $size_y$, заданные в виде одномерных векторов $P^{(l)} = P_1^{(l)}, \dots, P_{n_pix}^{(l)}$, дискретизированное начальное трехмерное изображение объекта реконструкции с размерностями по осям OX , OY и OZ — dim_x , dim_y и dim_z , представленное в виде одномерного вектора $V = (V_1, V_2, \dots, V_{pos_obj})$, число процессов n_pol , которое является делителем dim_y . А также $pos_obj = dim_x * dim_y * dim_z$; $n_pix = size_x * size_y$; $p_pol = pos_obj / n_pol$.

Выход. Для каждого вокселя V_j , $j = 1, \dots, pos_obj$, трехмерного цифрового изображения объекта рассчитано соответствующее значение линейного коэффициента ослабления μ_j внутри объекта реконструкции.

1. Инициализируем систему параллельного программирования MPI:

MPI_Init(& argc, & argv);

MPI_Comm_size(MPI_COMM_WORLD, & size);

MPI_Comm_rank(MPI_COMM_WORLD, & rank);

MPI_Get_processor_name(processor_name, & namelen).

2. Создаем топологию “полный граф” с p_pol независимо выполняющимися процессами *MPI_Graph_create(MPI_COMM_WORLD, p_pol, index, edges, reord, & comm_gr).*

3. Для всех процессов одновременно выполняем последовательность операций.

3.1. Для каждого процесса и соответствующей ему части изображения $V^{(n)}$, $n \in \{1, \dots, n_pol\}$, с числом вокселей p_pol строим свою проекционную матрицу:

for (i = 0; i <= n_prj; i++) InitPrjMatr(p_pol, i, rank, comm_gr).

3.2. Для каждого процесса организуем **цикл 1** по числу итераций:

for (k = 0; k < n_iter; k++) {Тело цикла 1}.

3.3. Внутри тела **цикла 1** для каждой части изображения $V^{(n)}$, $n \in \{1, \dots, n_pol\}$, организуем **цикл 2** по числу вокселей p_pol :

for (j = 0; j < p_pol; j++) {Тело цикла 2}.

3.4. Внутри тела **цикла 2** организуем **цикл 3** по числу проекций n_prj :

for (l = 0; l < n_prj; l++) {Тело цикла 3}.

3.5. Внутри тела **цикла 3** для каждого вокселя $V_j^{(n)}$, $j = 1, \dots, p_pol$, записываем в пиксели $P^{(l)}$ коэффициенты взаимодействия луча, проведенного из соответствующего пикселя к положению рентгеновского источника, с данным вокселем.

3.6. После окончания **цикла 2** и **цикла 3** в n_prj двумерных массивах $stp[l][n_pix]$, $l = 1, \dots, n_prj$, будут находиться рассчитанные лучевые суммы для соответствующей каждому процессу части изображения $V^{(n)}$, $n \in \{1, \dots, n_pol\}$, с числом вокселей p_pol . Для того чтобы все процессы синхронно закончили свои вычисления, останавливаемся и ждем, когда финиширует самый медленный процесс

MPI_Barrier(comm_gr).

3.7. Суммируем в массивах $smr[l]$ [n_pix] массивы $smp[l]$ [n_pix] для всех n_pol процессов и затем перезаписываем их обратно в массивы $smp[l]$ [n_pix]:

```
for(l = 0; l < n_prj; l++) MPI_Allreduce(smp[l],smr[l],n_pix,MPI_UNSIGNED_SHORT,
MPI_SUM, comm_gr);
```

```
for(l = 0; l < n_prj; l++) for(i = 0; i < n_pix; i++) smp[l][i]=smr[l][i].
```

3.8. Внутри тела **цикла 1** для каждой части изображения $V^{(n)}$, $n \in \{1, \dots, n_pol\}$, организуем **цикл 4** по числу вокселей p_pol :

```
for(j = 0; j < p_pol; j++) { Тело цикла 4 }.
```

3.9. Внутри тела **цикла 4** организуем **цикл 5** по числу проекций n_prj :

```
for(l = 0; l < n_prj; l++) { Тело цикла 5 }.
```

3.10. Внутри тела **цикла 5** для каждого вокселя $V^{(n)}_j$, $j = 1, \dots, p_pol$, рассчитываем поправку путем суммирования для всех пикселей $P^{(l)}$, в которые проецируется данный воксель, разности значений экспериментальной проекции $smf[l][i]$ и рассчитанной проекции $smp[l][i]$, $i \in \{1, \dots, n_pix\}$, умноженной на значение коэффициента взаимодействия луча, проведенного из данного пикселя к положению рентгеновского источника для данной проекции, с вокселем $V^{(n)}_j$:

```
objp[j]= (smf[l][i] - smp[l][i]) * dl_pr[l][i].
```

Вносим эту поправку, умноженную на значение коэффициента релаксации $lambda$, в одномерный массив $objr[j]$, $j = 1, \dots, p_pol$, представляющий собой вектор текущего значения коэффициента линейного ослабления μ_j для рассматриваемого процесса

```
objr[j]= objr[j]+ lambda * objp[j].
```

Для того чтобы все процессы синхронно закончили свои вычисления, останавливаемся и ждем, когда финиширует самый медленный процесс

```
MPI_Barrier(comm_gr).
```

3.11. Собираем скорректированные значения каждой части изображения $V^{(n)}$, $n \in \{1, \dots, n_pol\}$, с числом вокселей p_pol в полное изображение с числом вокселей pos_obj , расположенное в одномерном массиве $objf[pos_obj]$

```
MPI_Allgather(objr,p_pol,MPI_UNSIGNED_SHORT,objf,p_pol,
MPI_UNSIGNED_SHORT,comm_gr).
```

3.12. if Число итераций меньше n_iter

then.

Заканчиваем тело **цикла 1** для очередной итерации и переходим к следующей итерации

else.

Заканчиваем тело **цикла 1** и выходим из него

end if.

4. Заканчиваем работу с системой параллельного программирования MPI и выходим из программы

```
MPI_Finalize(); return(1).
```

2. Примеры томографической реконструкции изображений промышленных изделий

Исследования возможности создания в Республике Беларусь промышленных томографов проведены на основе уже выпускаемого серийно рентгеновского оборудования фирмы "Адани" (Минск). Выполнено рентгеновское сканирование пяти бескорпусных иголь-

чатых подшипников 664715 Д на рентгеновской установке “Пульмоскан-760У”, имеющей подвижный линейный детектор с 2560 элементами (рис. 1).

Физический размер детекторного элемента равнялся 0.2 мм, расстояние от рентгеновского источника до планки детекторов 1385 мм. Рабочее напряжение рентгеновской трубки 120 кВ, рабочий ток 25 мА, размер фокусного пятна 0.3 мм (X-RAY генератор Sedecal “SHF-335”, рентгеновская трубка VARIAN G-297 имеет максимальное напряжение 150 кВ, максимальный ток 150 мА). Получено 120 цифровых рентгеновских снимков размером 2560×2560 с разрешением 0.2×0.2 мм через интервал 3 град. для полного диапазона 360 град.

На рис. 2 показана рентгеновская проекция пяти сепараторов с интенсивностями непоглощенного излучения для начального угла поворота, равного 0 град. Рисунок приведен в горизонтальном положении, в реальном эксперименте вращение сепараторов осуществлялось вокруг вертикальной оси.

В результате итерационной трехмерной томографической реконструкции для сепараторов бескорпусных игольчатых подшипников 664715 Д получены трехмерные томограммы размером $580 \times 280 \times 580$ объемных элементов с разрешением $0.16 \times 0.16 \times 0.16$ мм.

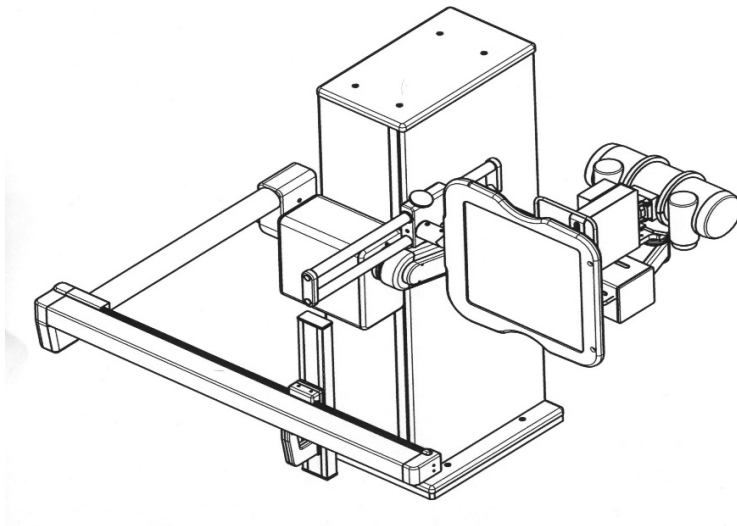


Рис. 1. Рентгеновская установка “Пульмоскан-760У” (вид спереди)

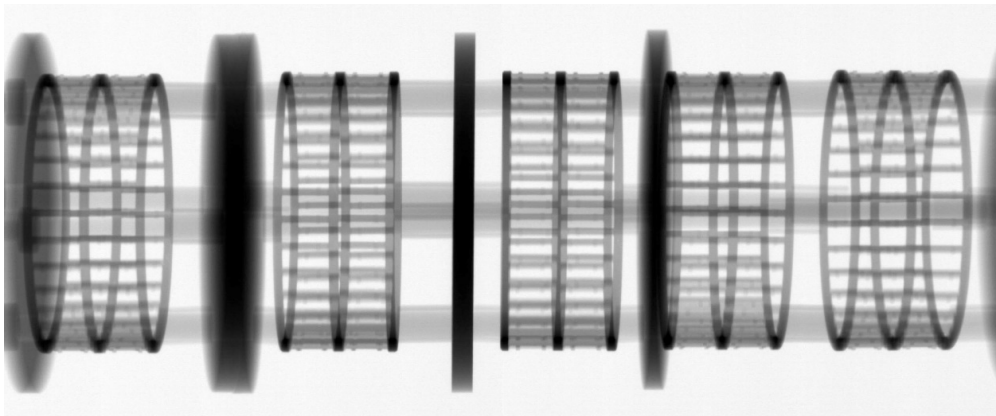


Рис. 2. Рентгеновская проекция пяти сепараторов для угла 0 град.

Время выполнения одной итерации 1 мин. Реконструкция осуществлена как по 60 проекциям с угловым шагом между проекциями 6 град., так и по 120 проекциям через угловой шаг между проекциями, равный 3 град.

На рис. 3 приведено изображение одного из реконструированных сепараторов в трех различных пространственных положениях, а на рис. 4 представлены изображения сечений сепаратора плоскостями XOY и YOZ в местах, где предполагается наличие дефектов (пор).

Кроме того, проведено рентгеновское сканирование вентилятора и масляного распределителя на рентгеновской установке фирмы “Адани” Securescan, имеющей неподвижный Г-образный детектор с 3136 регистрирующими элементами размером 0.8 мм (детали перемещались относительно планки детекторов на подвижной платформе). Расстояние от рентгеновского источника до детектора 1799 мм, рабочее напряжение трубки 160 кВ, размер фокусного пятна 0.8 мм. Получено 60 цифровых рентгеновских снимков вентилятора размерами 1150×3136 с разрешением 0.6×0.8 мм в угловом диапазоне 180 град. с шагом 3 град. и 60 снимков для масляного распределителя. На рис. 5 показан

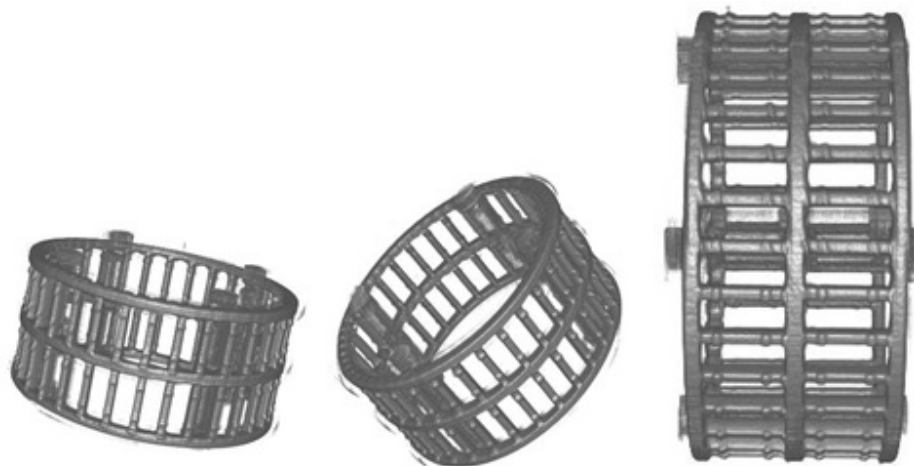


Рис. 3. Реконструированное изображение сепаратора в трех положениях

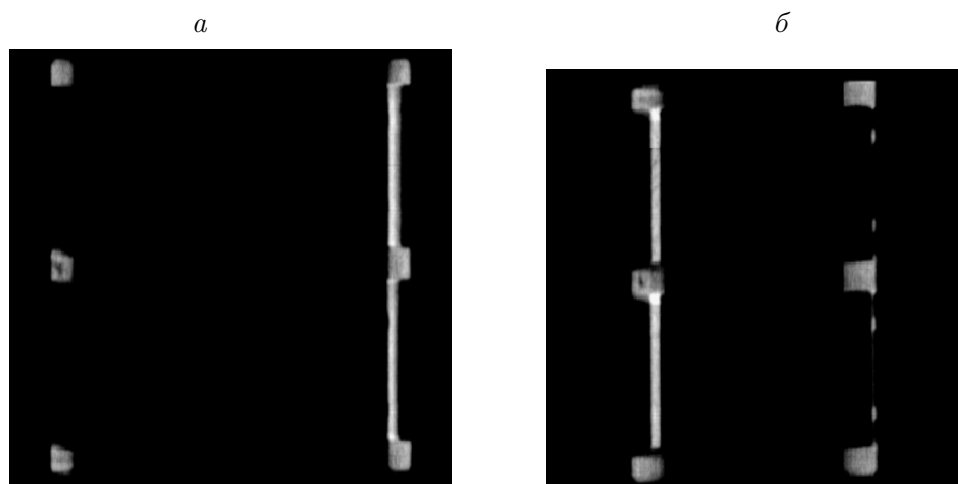


Рис. 4. Сечения сепаратора плоскостью XOY (а) и плоскостью YOZ , повернутое на 90° по часовой стрелке (б)

вид этой установки. Реконструировано трехмерное изображение вентилятора с разрешением $1 \times 1 \times 1$ мм, три вида этого изображения показаны на рис. 6.

Особый интерес представляет реконструкция изображения масляного распределителя. Его габаритные размеры составляют $134 \times 90 \times 45$ мм, а для напряжения 120 кВ, при котором производилась рентгеновская съемка, можно пробить слой стали поряд-

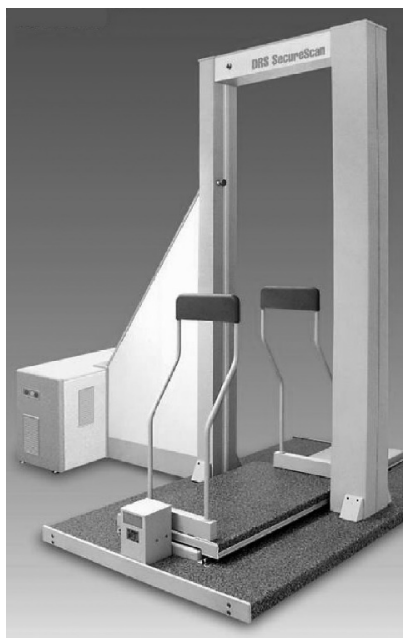


Рис. 5. Вид рентгеновской установки Securescan



Рис. 6. Реконструированные изображения вентилятора



Рис. 7. Реконструированное изображение масляного распределителя в трех положениях

ка 40 мм, т. е. на большинстве рентгеновских проекций имелись значительные потери информации. Но тем не менее благодаря коррекции эффекта ужесточения лучей и учета дополнительной априорной информации о реальном значении коэффициента линейного ослабления стали для данной энергии просвечивания удалось в принципе восстановить внутреннюю структуру масляных каналов. На рис. 7 приведено изображение масляного распределителя в трех пространственных положениях.

Для выполнения расчетов были использованы вычислительные ресурсы ЦКП “Центр данных ДВО РАН” [11].

Таким образом, разработанная параллельная итерационная технология реконструкции трехмерных изображений имеет бесспорные преимущества по сравнению с традиционной последовательной итерационной томографической реконструкцией. Она позволяет в десятки раз сократить время осуществления томографической реконструкции, обеспечивает возможность восстановления изделий размерностями $512^3 - 1024^3$ вокселей с одновременным хранением в оперативной памяти узла подматрицы проекционной матрицы, что освобождает от необходимости пересчета матричных коэффициентов для каждой новой итерации.

Список литературы / References

- [1] **Абламейко С.В., Абрамов С.М.** Основные результаты суперкомпьютерной программы “СКИФ” Союзного Государства // Тр. семинара АКИИ’03: Третий расширенный семинар “Использование методов искусственного интеллекта и высокопроизводительных вычислениях и в аэрокосмических исследованиях”. Переславль-Залесский: ИПС РАН, 2003. С. 135–140.
Albameyko, S.V., Abramov, S.M. Main results of the Union State “SKIF” supercomputer program // Tr. Seminara AKII’03: Tretiy Rasshirennyy Seminar “Ispol’zovanie Metodov Iskusstvennogo Intellekta i Vysokoproizvoditel’nykh Vychisleniyakh i v Aerokosmicheskikh Issledovaniyakh”. Pereslavl’-Zalesskiy: IPS RAN, 2003. P. 135–140. (In Russ.)
- [2] **Фаддеева В.Н., Фаддеев Д.К.** Параллельные вычисления в линейной алгебре. Ч. 1 // Кибернетика. 1977. № 6. С. 28–40.
Faddeeva, V.N., Faddeev, D.K. Parallel computing in linear algebra. Pt 1 // Cybernetics. 1977. No. 6. P. 28–40. (In Russ.)
- [3] **Фаддеева В.Н., Фаддеев Д.К.** Параллельные вычисления в линейной алгебре. Ч. 2 // Кибернетика. 1982. № 3. С. 18–31.
Faddeeva, V.N., Faddeev, D.K. Parallel computing in linear algebra. Pt 2 // Cybernetics. 1982. No. 3. P. 18–31. (In Russ.)
- [4] **Фаддеев Д.К., Фаддеева В.Н.** Вычислительные методы линейной алгебры. СПб.: Лань, 2002. 736 с.
Faddeev, D.K., Faddeeva, V.N. Parallel computing in linear algebra. St. Petersburg: Lan’, 2002. 736 p. (In Russ.)
- [5] **Воеводин В.В.** Математические модели и методы в параллельных процессах. М.: Наука, 1986. 296 с.
Voevodin, V.V. Mathematical models and methods in parallel processes. Moscow: Nauka, 1986. 296 p. (In Russ.)
- [6] **Воеводин В.В.** Параллельные структуры алгоритмов и программ. М.: ОВМ АН СССР, 1987. 148 с.
Voevodin, V.V. Parallel structures of algorithms and programs. Moscow: OVM AN SSSR, 1987. 148 p. (In Russ.)

- [7] **Воеводин В.В.** Параллельные вычисления. СПб.: БХВ–Петербург, 2004. 608 с.
Voevodin, V.V. Parallel computations. St. Petersburg: BKhV-Petersburg, 2004. 608 p.
(In Russ.)
- [8] **Ортега Дж.** Введение в параллельные и векторные методы решения линейных систем. М.: Мир, 1991. 367 с.
Ortega, J.M. Introduction to parallel and vector solution of linear systems. Moscow: Mir, 1991. 367 p. (In Russ.)
- [9] **Bertsekas, D.P.** Parallel and distributed computation: Numerical methods. NJ: Prentice-Hall, Upper Saddle River, 1989. 716 p.
- [10] **Корнеев В.Д.** Параллельное программирование в MPI. М.; Ижевск: Институт компьютерных исследований, 2003. 304 с.
Korneev, V.D. Parallel programming. Moscow; Izhevsk: Institute of Computer Research, 2003. 304 p. (In Russ.)
- [11] **Сорокин А.А., Макогонов С.В., Королев С.П.** Информационная инфраструктура для коллективной работы ученых Дальнего Востока России // Научно-техническая информация. Сер. 1: Организация и методика информационной работы. 2017. № 12. С. 4–16.
Sorokin, A.A., Makogonov, S.V., Korolev, S.P. Information infrastructure for scientists of the Russian Far East region collective work // Nauchno-Tekhnicheskaya Informatsiya. Ser. 1: Organizatsiya i Metodika Informatsionnoy Raboty. 2017. No. 12. P. 14–16. (In Russ.)

Поступила в редакцию 21 октября 2019 г.

Synchronous algorithm for parallel reconstruction of three-dimensional images based on the MPI system

ZOLOTAREV, SERGEY A.¹, VENGRINOVICH, VALERY L.¹, SMAGIN, SERGEY I.^{2,*}

¹Institute of Applied Physics of NASB, Minsk, 220072, Belarus

²Computing Center of FEB RAS, Khabarovsk, 680000, Russia

*Corresponding author: Smagin, Sergey I., e-mail: smagin@ccfebras.ru

Purpose. Currently, one of the most important tasks is the development and adaptation of iterative methods for solving ultra-large sparse systems of algebraic equations. Such computational problems are caused by the iterative parallel reconstruction of three-dimensional images of industrial products. It is important that iterative methods for solving computational problems of large size are implemented on parallel structures much more efficiently than direct methods for solving them. This paper describes a synchronous parallel algorithm based on the MPI system for solving the problem of reconstruction of three-dimensional images of industrial products.

Methodology. It is important that iterative methods for solving computational problems of larger dimensionality on parallel structures are implemented more efficiently than the direct ones. Most algorithms based on direct solving methods have a significant hereditary sequential structure and require a large number of processors interactions which cannot be executed in parallel mode. Iterative methods, for the most part, require a significantly smaller number of interactions of this type and are relatively easily mapped onto parallel computational structures. Equally important, in most cases parallel implementations of classical iterative methods are more effective in terms of

computational speed. The parallel execution of the algorithm is based on the distribution of the process in some way between various groups of processors. Depending on the interaction method between local processors, two different types of parallel iterative algorithms execution are distinguished: synchronous and asynchronous. In the former case, it is assumed that the processors complete the calculations and exchange all the necessary results before the start of a new iteration. The main disadvantage of synchronous parallel algorithms is that they require synchronization of iterations. This is a very difficult task, especially with large number of processors. In addition, the overall calculation speed is limited by the speed of the slowest processor. At the same time, faster processors spend most of their time in the waiting mode. But, nevertheless, the implementation of these parallel algorithms can be effectively achieved using the MPI standard.

Findings. Synchronous parallel computing algorithms and program codes for three-dimensional tomographic reconstruction in a conical beam were developed. Program code debugging and numerical calculations were performed on a hybrid cluster based on the OpenPOWER architecture using the MPI system. For designing a parallel three-dimensional tomographic reconstruction, a voxel form of parallelism was used.

Originality. Implemented parallel iterative technology of three-dimensional images reconstruction has undeniable advantages over traditional sequential iterative tomographic reconstruction. It allows reducing the time of tomographic reconstruction as many as tens of times, provides the ability to reconstruct products with sizes of 5123 to 10243 voxels with simultaneous storage of the submatrix node of the projection matrix in RAM, which eliminates the need for recalculation of matrix coefficients for each new iteration.

Keywords: iterative methods, image reconstruction, tomography, parallel algorithms, MPI, voxel concurrency.

Cite: Zolotarev S.A., Vengrinovich V.L., Smagin S.I. Synchronous algorithm for parallel reconstruction of three-dimensional images based on the MPI system // Computational Technologies. 2019. Vol. 24, No. 6. P. 30–39. DOI: 10.25743/ICT.2019.24.6.005. (In Russ.)

Received October 21, 2019