

Декомпозиция области на основе прямого метода решения трехмерного уравнения Пуассона в нестационарных задачах астрофизики*

Н.В. Снытников¹

Институт Вычислительной Математики и Математической Геофизики СО РАН¹

Предложен новый параллельный алгоритм для решения трехмерного уравнения Пуассона в нестационарных задачах астрофизики. Алгоритм основан на декомпозиции области по двум направлениям, применении прямого метода решения задачи Дирихле в каждой подобласти и комбинации метода сопряжения подобластей для двумерного экранированного уравнения Пуассона с методом разделения переменных.

Тестовые эксперименты были проведены на суперкомпьютерах Межведомственного суперкомпьютерного центра (МСКЦ) и Сибирского суперкомпьютерного центра (ССКЦ) и продемонстрировали хорошую масштабируемость метода до тысячи процессоров и его перспективность для использования с несколькими тысячами процессоров.

1. Введение

Для некоторых задач астрофизики, таких как изучение динамики галактик или околозвездных дисков, возникает необходимость решать систему дифференциальных уравнений, описывающих нестационарную динамику вещества в гравитационном поле [1]. Такие системы, как правило, решаются численно на основе методов расщепления по физическим процессам и требуют многократного решения трехмерного уравнения Пуассона для гравитационного потенциала. Поскольку общее количество шагов может составлять десятки тысяч, а размеры сетки при этом достигают миллиардов узлов (то есть порядка 10^{24} и более), то необходимо создавать эффективные параллельные алгоритмы, предназначенные для суперкомпьютеров с большим количеством процессоров.

Ранее, в статье [2], был предложен алгоритм для решения двумерного уравнения Пуассона, основанный на сопряжении смежных подобластей с помощью вычисления потенциала граничного слоя [3] и использовании нестационарности исходной задачи для предвычисления вспомогательных величин в Фурье-разложении потенциала граничного слоя, что идеологически близко к подходу [4]. В настоящей работе мы обобщаем этот метод для применения к трехмерному случаю: выполняется декомпозиция трехмерной области по двум направлениям, применяется метод разделения переменных (преобразование Фурье) по одному направлению и решается набор задач Дирихле для экранированного уравнения Пуассона.

Статья имеет следующую структуру. В разделе 2 представлено схематичное описание разработанного алгоритма для двумерного экранированного уравнения Пуассона (более подробное описание может быть найдено в работе [2]). Раздел 3 посвящен описанию алгоритмов декомпозиции трехмерной области в одном и двух направлениях. В разделе 4 приведены результаты измерения производительности созданных программных реализаций, в разделе 5 — выводы.

*Работа выполнена при поддержке Российского Научного Фонда, грант № 14-11-00485.

2. Параллельный метод решения двумерного экранированного уравнения Пуассона

Пусть задана прямоугольная двумерная область Ω с границей Γ . Требуется решить задачу Дирихле для уравнения Пуассона (1) в предположении, что она должна решаться многократно для разных функций ρ :

$$\Delta\Phi(\mathbf{x}) - a^2\Phi(\mathbf{x}) = \rho(\mathbf{x}), \quad \Phi(\mathbf{x})|_{\Gamma} = 0, \quad (1)$$

где $\mathbf{x} = (x, y)$.

Введем равномерную сетку с количеством узлов $L_x \times L_y$ и пространственными шагами h_x, h_y . Для индексации узлов сетки будем использовать $i = 0, \dots, L_x, k = 0, \dots, L_y$. Для удобства изложения ограничимся аппроксимацией оператора Лапласа с помощью стандартного 5-точечного шаблона и предполагаем, что заданы однородные краевые условия для задачи Дирихле. Вместе с тем, ничто не мешает рассматривать другой компактный шаблон аппроксимации и неоднородную задачу, которая может быть сведена к однородной стандартным способом [5].

Тогда разностная задача будет записана в следующем виде:

$$\frac{\Phi_{i+1,k} - 2\Phi_{i,k} + \Phi_{i-1,k}}{h_x^2} + \frac{\Phi_{i,k+1} - 2\Phi_{i,k} + \Phi_{i,k-1}}{h_y^2} - a^2\Phi_{i,k} = \rho_{ik}, \quad (2)$$

$$\Phi|_{\Gamma} = 0.$$

2.1. Вычисление потенциала слоя зарядов

Определим *слой сетки с индексом i_0* как набор сеточных узлов (i_0, k) , имеющих фиксированный индекс i_0 и $k = 0, \dots, L_y$. Рассмотрим следующую подзадачу для задачи (10): предполагаем, что плотность ρ равна нулю во всей сеточной области кроме единственного слоя с индексом i_0 : $\rho_{i_0k} \neq 0$. Требуется найти значения потенциала Φ_{i_1k} на слое i_1 (см. Рис. 1).

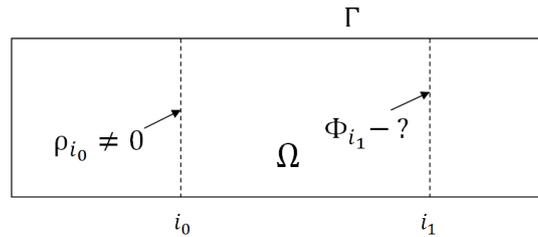


Рис. 1. Задача вычисления потенциала на слое с индексом i_1 , генерируемого плотностью ρ_{i_0k} , локализованной на слое i_0 .

Оказывается, что при определенных условиях вычислительную трудоемкость решения такой подзадачи можно существенно уменьшить: вплоть до $O(L_x \log L_y)$. Это намного лучше, чем решать задачу во всей области, применяя, например, прямой метод разделения переменных, сложность которого составляет $O(L_x L_y (\log L_x + \log L_y))$.

Дело в том, что раскладывая сеточную функцию потенциала в ряд Фурье по синусам, после некоторых преобразований можно получить следующую формулу:

$$\begin{aligned} \Phi_{i_1k} &= \frac{1}{2L_y} \sum_{n=1}^{L_y-1} F_{i_1}(n) \sin \frac{\pi nk}{L_y} \\ &= \frac{1}{2L_x} \frac{1}{2L_y} \sum_{n=1}^{L_y-1} A_{i_0}(n) \cdot B_{i_1, i_0}(n) \cdot \sin \frac{\pi nk}{L_y}, \end{aligned} \quad (3)$$

где:

$$\begin{aligned}
 A_{i_0}(n) &= \sum_{k=1}^{L_y-1} \rho_{i_0 k} \sin \frac{\pi n k}{L_y}, \\
 B_{i,i_0}(n) &= \sum_{m=1}^{L_x-1} S(m, n) \sin \frac{\pi m i_0}{L_x} \sin \frac{\pi m i}{L_x}, \\
 S(m, n) &= \left[-\frac{4 \sin^2(\frac{\pi m}{2L_x})}{h_x^2} - \frac{4 \sin^2(\frac{\pi n}{2L_y})}{h_y^2} - a^2 \right]^{-1}.
 \end{aligned} \tag{4}$$

Поскольку величины $B_{i,i_0}(n)$ зависят только от области решения, параметров сетки и схемы аппроксимации, но не зависят от функции ρ , то они могут быть посчитаны заранее для заданной сетки и пары заданных значений i, i_0 . Сложность их вычисления составляет $O(L_x L_y)$, а объем памяти, требуемый для хранения $B_{i,i_0}(n)$, незначителен по сравнению с объемом памяти для всей задачи – порядка L_y значений. Таким образом, трудоемкость вычисления $\Phi_{i_0 k}$ существенно уменьшается и составляет $O(L_y \log L_y)$.

2.2. Декомпозиция двумерной области на две подобласти

Данный алгоритм вычисления потенциала, создаваемого слоем зарядов, можно применить для параллельного решения задачи (2). Для этого рассмотрим задачу декомпозиции области Ω на две подобласти Ω_1, Ω_2 с границами Γ_1, Γ_2 и общей границей γ (Рис. 2).

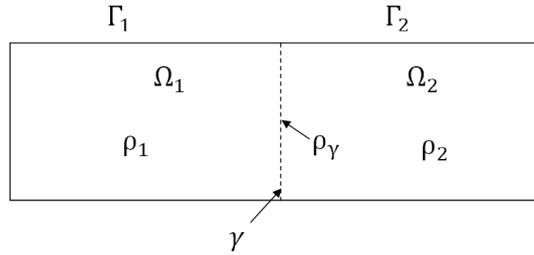


Рис. 2. Подразделение двумерной области на две подобласти.

Запишем задачу Дирихле для экранированного уравнения Пуассона в каждой подобласти, считая, что значение потенциала на общей границе γ равно нулю, а сеточные функции ρ_1, ρ_2 совпадают с сеточной функцией ρ в своих подобластях:

$$\begin{aligned}
 \Delta \Phi_1 - a^2 \Phi_1 &= \rho_1, \quad \Phi_1|_{\Gamma_1} = 0, \\
 \Delta \Phi_2 - a^2 \Phi_2 &= \rho_2, \quad \Phi_2|_{\Gamma_2} = 0.
 \end{aligned} \tag{5}$$

Решим эти задачи для обеих подобластей и объединим полученные решения Φ_1 и Φ_2 :

$$\Phi_0(\mathbf{x}) = \begin{cases} \Phi_1(\mathbf{x}), & \mathbf{x} \in \Omega_1, \\ \Phi_2(\mathbf{x}), & \mathbf{x} \in \Omega_2, \\ 0, & \mathbf{x} \in \gamma \end{cases}$$

Очевидно, $\Phi_0(\mathbf{x})$ не будет удовлетворять исходной задаче (2). Во-первых, не учитывается ненулевой слой исходной плотности на границе подобластей γ (обозначим его ρ_γ). И, во-вторых, функция Φ_0 , являясь непрерывной в области Ω , имеет разрыв нормальной производной при пересечении границы γ :

$$\frac{\partial \Phi_1}{\partial \vec{n}}|_\gamma \neq -\frac{\partial \Phi_2}{\partial \vec{n}}|_\gamma$$

Этот разрыв производной порождает так называемый слой экранирующих зарядов q^{scr} , плотность которых может быть вычислена с помощью применения разностного оператора Лапласа к значениям потенциала на границе:

$$\rho^{scr} := \Delta\Phi_0(\mathbf{x}) - a^2\Phi_0(\mathbf{x}), \quad \mathbf{x} \in \gamma.$$

Предположим теперь, что мы вычислили потенциал Φ^{scr} во всей области Ω , создаваемый плотностью $\rho_\gamma - \rho^{scr}$ и являющийся решением задачи:

$$\Delta\Phi^{scr} - a^2\Phi^{scr} = \rho_\gamma - \rho^{scr}, \quad \Phi^{scr}|_\Gamma = 0 \quad (6)$$

Тогда сумма решений $\Phi = \Phi_0 + \Phi^{scr}$ будет являться исходным решением задачи (2). Действительно:

$$\begin{aligned} \Delta\Phi(\mathbf{x}) - a^2\Phi(\mathbf{x}) &= \Delta\Phi_0(\mathbf{x}) - a^2\Phi_0(\mathbf{x}) + \Delta\Phi^{scr}(\mathbf{x}) - a^2\Phi^{scr}(\mathbf{x}) \\ &= \begin{cases} \Delta\Phi_1(\mathbf{x}) - a^2\Phi_1(\mathbf{x}) = \rho_1(\mathbf{x}) = \rho(\mathbf{x}), & \mathbf{x} \in \Omega_1, \\ \Delta\Phi_2(\mathbf{x}) - a^2\Phi_2(\mathbf{x}) = \rho_2(\mathbf{x}) = \rho(\mathbf{x}), & \mathbf{x} \in \Omega_2, \\ \rho^{scr} + \rho_\gamma - \rho^{scr} = \rho_\gamma \equiv \rho(\mathbf{x}), & \mathbf{x} \in \gamma. \end{cases} \end{aligned}$$

Также потенциал Φ по своему построению удовлетворяет краевым условиям задачи (2).

Подобный подход к сопряжению подобластей на основе объединения общего решения уравнения Пуассона и частного решения уравнения Лапласа хорошо известен, и в том или ином виде он использовался во многих методах [3, 6–8]. Новым в данной работе является применение более эффективного метода для решения задачи (6), использующего нестационарность исходной задачи.

Ведь, действительно, с помощью описанного выше алгоритма можно быстро вычислить потенциал Φ_γ^{scr} на слое γ , который создается зарядами с плотностью $\rho_\gamma - \rho^{scr}$, расположенными на том же слое. Далее, решая независимо задачи Дирихле для экранированного уравнения Лапласа в каждой подобласти, найдем $\Phi_1^{scr}, \Phi_2^{scr}$:

$$\begin{aligned} \Delta\Phi_1^{scr} - a^2\Phi_1^{scr} &= 0, \quad \Phi_1|_{\Gamma_1 \setminus \gamma} = 0, \quad \Phi_1|_\gamma = \Phi_\gamma^{scr} \\ \Delta\Phi_2^{scr} - a^2\Phi_2^{scr} &= 0, \quad \Phi_2|_{\Gamma_2 \setminus \gamma} = 0, \quad \Phi_2|_\gamma = \Phi_\gamma^{scr} \end{aligned} \quad (7)$$

Накладными расходами этого алгоритма декомпозиции, возникающими в дополнение к вычислительной трудоемкости последовательного решения задачи (2), являются необходимость двукратного решения задачи Дирихле (для уравнения Пуассона и уравнения Лапласа) в каждой подобласти, передача массива размера L_y между процессорами и вычисление потенциала Φ_γ^{scr} . По сути это означает, что за счет двукратного увеличения вычислительной нагрузки удается снизить объем межпроцессорных коммуникаций. Также заметим, что для решения СЛАУ, возникающих в задачах (5) и (7), можно применять любые прямые методы.

2.3. Декомпозиция двумерной области на N подобластей

Теперь предположим, что область Ω разделена в одном направлении на N подобластей Ω_n , $n = 0, \dots, N - 1$, каждая из которых назначена для обработки своему собственному процессору. Далее мы будем предполагать, что количество подобластей является степенью двойки $N = 2^{N_i}$.

Для того, чтобы эффективно вычислить потенциал, создаваемый зарядами, находящимися на границах подобластей, был разработан алгоритм, основанный на построении

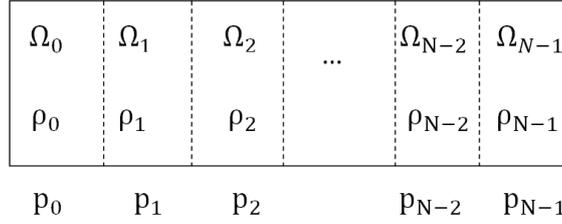


Рис. 3. Подразделение 2D области на N подобластей в направлении x .

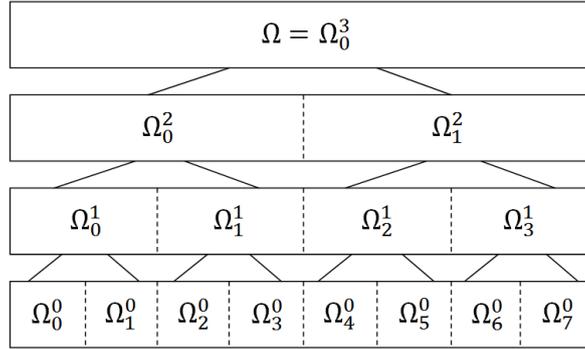


Рис. 4. Построение иерархии подобластей ($N = 8$) в виде двоичного дерева.

иерархии подобластей в виде двоичного (бинарного) дерева. Общая схема дерева для $N = 8$ представлена на Рис. 4.

Корнем дерева является вся область решения Ω . Будем обозначать подобласти решения $\Omega = \Omega_0^{N_l-1}$, где верхний индекс является порядковым номером уровня узла в дереве, или его глубиной, (нумерация уровней в дереве начинается с листьев), а нижний индекс обозначает порядковый номер подобласти на данном уровне. Дочерними элементами корня дерева являются две одинаковых подобласти $\Omega_0^{N_l-2}, \Omega_1^{N_l-2}$, а их дочерними элементами, соответственно: $\Omega_0^{N_l-3}, \Omega_1^{N_l-3}, \Omega_2^{N_l-3}, \Omega_3^{N_l-3}$. Построение дерева продолжается до того момента, пока все N подобластей Ω_n не станут листьями в дереве. Они будут обозначаться $\Omega_n^0, n = 0, \dots, N-1$.

Вычисления начинаются с листьев дерева: в каждой подобласти решается уравнение Пуассона, а затем происходит объединение двух подобластей и переход на следующий уровень дерева. На нем объединяются следующие подобласти, и процесс продолжается вплоть до достижения корня дерева. Делая шаг по дереву, каждый процессор выполняет коммуникации с одним и только одним процессором для передачи необходимых значений для объединения двух подобластей, а также вычисляет значение потенциала на границе исходной подобласти (листового узла). В итоге на каждом процессоре будут все необходимые данные для решения уравнения Лапласа и конструирования итогового решения. И каждому процессору необходимо сделать $N_l = \log_2 N$ шагов вверх по дереву и на каждом шаге выполнить передачу и получение массива данных размером $3L_y$. Соответственно, если для решения уравнения Пуассона и Лапласа применять двукратное быстрое преобразование Фурье, то вычислительная сложность алгоритма декомпозиции составит:

$$T_{calc}(L_x, L_y, N) = O\left(\frac{L_x L_y (\log_2 \frac{L_x}{N} + \log L_y)}{N}\right) + O(L_y \log L_y \log N). \quad (8)$$

Первый член в формуле отражает трудоемкость решения уравнения Пуассона и Лаплас для каждой подобласти, а второй член отвечает за вычисления влияния слоя зарядов для постановки граничных условий.

Коммуникационная сложность для каждого процессора (то есть тот объем данных, которые необходимо отправить и получить каждому процессору) составляет:

$$T_{comm}(L_x, L_y, N) = (\text{Send}(3L_y) + \text{Recv}(3L_y)) \log_2 N. \quad (9)$$

Таким образом, если $N \ll L_x$ (например, $N \leq 32 \cdot L_x$), объем коммуникаций относительно невелик в сравнении с объемом вычислительной задачи, решаемой на каждом процессоре.

3. Параллельный метод решения трехмерного уравнения Пуассона

Перейдем теперь к описанию метода декомпозиции для трехмерной прямоугольной области Ω с границей Γ . Требуется решить задачу Дирихле для уравнения Пуассона (10) в предположении, что она должна решаться многократно для разных функций ρ :

$$\Delta\Phi(\mathbf{x}) = \rho(\mathbf{x}), \quad \Phi(\mathbf{x})|_{\Gamma} = 0, \quad (10)$$

где $\mathbf{x} = (x, y, z)$.

Аналогично двумерному случаю введем равномерную сетку с количеством узлов $L_x \times L_y \times L_z$ и пространственными шагами h_x, h_y, h_z . Для индексации узлов сетки будем использовать $i = 0, \dots, L_x, j = 0, \dots, L_y, k = 0, \dots, L_z$. Аппроксимируя оператор Лапласа с помощью стандартного 7-точечного шаблона и применяя метод разделения переменных (разложение в ряд Фурье по синусам) в направлении y , получим следующую серию независимых систем уравнений:

$$\frac{\Phi_{i+1,k}(p) - 2\Phi_{i,k}(p) + \Phi_{i-1,k}(p)}{h_x^2} + \frac{\Phi_{i,k+1}(p) - 2\Phi_{i,k}(p) + \Phi_{i,k-1}(p)}{h_z^2} - a^2\Phi_{i,k}(p) = \rho_{ik}(p),$$

$$p = 1, \dots, L_y - 1, \quad (11)$$

где

$$a^2 = \frac{4 \sin^2\left(\frac{\pi l}{2L_y}\right)}{h_y^2}.$$

Каждое из этих уравнений является разностным аналогом для двумерного экранированного уравнения Пуассона и может быть параллельно решено с помощью описанного выше алгоритма. Таким образом, можно сформулировать следующий алгоритм для декомпозиции трехмерной области по одному направлению.

Алгоритм 1: декомпозиция трехмерной области по одному направлению.

1. Область решения подразделяется на $N = N_x$ подобластей по одному направлению x . Каждой подобласти Ω_n назначается свой процессор P_n .
2. В каждой подобласти Ω_n выполняется преобразование Фурье (разложение в ряд по синусам) в направлении y .
3. Параллельно решаются задачи (11) методом декомпозиции для двумерного экранированного уравнения Пуассона.
4. В каждой подобласти Ω_n выполняется обратное преобразование Фурье в направлении y .
5. В итоге на каждом процессоре будет получена соответствующая его подобласти Ω_n сеточная функция потенциала.

Ясно, что такой алгоритм может быть применен только в том случае, когда $N_x \ll L_x$ (например, $N_x = 32, L_x = 1024$). Однако для сеток порядка 1024^3 использование 32 процессоров будет недостаточно, поэтому необходима декомпозиция в двух направлениях. С одной стороны, вследствие независимости систем уравнений (11) можно применить метод декомпозиции на основе транспозиции [9] в направлении y и z . Вместе с тем, подобный метод будет наследовать часть проблем, присущих транспонированию данных, связанных с большим объемом межпроцессорных коммуникаций.

Поэтому более перспективной альтернативой является двукратное применение Алгоритма 1. Это позволит сократить объем межпроцессорных коммуникаций.

Алгоритм 2: декомпозиция трехмерной области по двум направлениям.

1. Область решения подразделяется на $N = N_x \times N_y$ одинаковых подобластей по двум направлениям x и y . Каждой подобласти назначается свой процессор.
2. В каждой подобласти решаем однородную задачу Дирихле для уравнения Пуассона и применяем Алгоритм 1 к объединению этих подобластей в направлении y .
3. Вычисляем величину экранирующих зарядов на границе подобластей в направлении x .
4. Применяем Алгоритм 1 по направлению x для вычисления потенциала на границах подобластей.
5. В каждой подобласти решаем задачу Дирихле для уравнения Лапласа с краевыми условиями, полученными на предыдущем шаге.
6. Итоговое решение будет получено как сумма сеточного решения уравнения Лапласа и решения уравнения Пуассона.

4. Тестовые эксперименты

Тестовые эксперименты проводились на двух суперкомпьютерах: на МВС-100К в Межведомственном суперкомпьютерном центре (четырёхъядерные процессоры Intel Xeon E5450 3 ГГц, с использованием MVARCH-1.2, компиляторов Intel C++ 12) и на суперкомпьютере в Сибирском суперкомпьютерном центре (четырёхъядерные процессоры Intel Xeon E5540 2.53 ГГц, с использованием Intel MPI 4.1 и компиляторов Intel C++ 14). Для выполнения быстрого преобразования Фурье использовалась библиотека FFTW 3.1.4 [10]. Далее приводятся только те результаты, которые получены в МСКЦ.

При замерах производительности время решения Алгоритма 2 разбивалось на три части:

- суммарное время решения уравнения Пуассона и Лапласа (T_{calc}),
- коммуникационная часть: передача массивов данных между процессорами с помощью MPI процедуры MPI_SendRecv и транспозиции граничных значений между подобластями (T_{comm}),
- вычисление потенциала выделенного слоя и граничных условий (T_{prop}).

В таблице 1 приведены результаты расчетов для сеток размера 1024^3 и 2048^3 с разным числом процессоров. Видно, что при увеличении количества процессоров от 64 до 1024 и сохранении размеров задачи (оценке так называемой сильной масштабируемости) коммуникационные расходы возрастают в процентном выражении с 7% до 25%. С другой стороны, сопоставив результаты запуска для 1024 процессоров на сетке 1024^3 и для 1024 процессоров на сетке 2048^3 , видно, что коммуникационные расходы для большей задачи уменьшаются до 15%.

Таблица 1. Эксперименты по оценке производительности алгоритма для сетки 1024^3 и 2048^3 при разном количестве процессоров. T_{all} – общее время счета, T_{comm} – коммуникационные расходы, T_{calc} – время однопроцессорных вычислений, T_{prop} – время вычисления значений потенциала на границе подобластей.

Число процессоров и размеры сетки		Время решения (секунды)			
$N = N_x \times N_y$	$L_x \times L_y \times L_z$	T_{all}	T_{calc}	T_{comm}	T_{prop}
$64 = 8 \times 8$	$1024 \times 1024 \times 1024$	8.33	7.34	0.48	0.50
$256 = 16 \times 16$	$1024 \times 1024 \times 1024$	2.32	1.72	0.27	0.32
$1024 = 32 \times 32$	$1024 \times 1024 \times 1024$	0.82	0.43	0.18	0.20
$1024 = 32 \times 32$	$2048 \times 2048 \times 2048$	5.24	3.64	0.79	0.81

5. Заключение

Был разработан и реализован параллельный алгоритм для решения трехмерного уравнения Пуассона для нестационарных задач. Его основным преимуществом является относительно небольшой объем межпроцессорных коммуникаций. Тестовые эксперименты демонстрируют приемлемую производительность алгоритма для расчетов до 1024 процессоров. Теоретические оценки показали перспективность использования для нескольких тысяч процессоров.

Литература

1. Снытников В.Н., Вшивков В.А., Кукшева Э.А., Неупокоев Е.В., Никитин С.А., Снытников А.В. Трехмерное численное моделирование нестационарной гравитирующей системы многих тел с газом // Письма в астрономический журнал. 2004. 30, N.2. 146-160.
2. Снытников Н.В. Параллельный алгоритм для решения 2D-уравнения Пуассона в контексте нестационарных задач // Вычислительные методы и программирование. 2015. 16. С.39-51.
3. Huang J., Greengard L. A Fast Direct Solver for Elliptic Partial Differential Equations On Adaptively Refined Meshes // SIAM J. Sci. Comput. 2000. Vol.21. P.1551–1566.
4. Terekhov A.V. Parallel Dichotomy Algorithm for Solving Tridiagonal System of Linear Equations with Multiple Right-Hand Sides // Parallel Computing. 2010. Vol.36. P.423–438.
5. Самарский А.А. Николаев Е.С. Методы решений сеточных уравнений. М.: Наука. 1978. 592 с.
6. James R.A. The Solution of Poisson's Equation for Isolated Source Distributions // Journal of Computational Physics, Vol.25, p.71, 1977.
7. Balls G.T., Colella P. A Finite Difference Domain Decomposition Method Using Local Corrections for the Solution of Poisson's Equation // J. Comp. Physics. 2002. Vol.180. P.25-53.

8. Яненко Н.Н., Коновалов А.Н., Бугров А.Н., Шустов Г.В. Об организации параллельных вычислений и «распараллеливании» прогонки // Численные методы механики сплошной среды. 1978. Т.9. С.139–146.
9. O. Ayala, L.P. Wang. Parallel implementation and scalability analysis of 3D Fast Fourier Transform using 2D domain decomposition // Parallel Computing. 2013. Vol.39. P. 58-77
10. M.Frigo and S.G.Johnson. FFTW software // <http://www.fftw.org>