

ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ ФУНДАМЕНТАЛЬНОГО РЕШЕНИЯ УРАВНЕНИЯ ПУАССОНА*

Э. А. КУКШЕВА, В. Н. СНЫТНИКОВ

Институт катализа им. Г. К. Борескова, Новосибирск, Россия

e-mail: kuksheva@catalysis.nsk.su, snyt@catalysis.nsk.su

A discrete analogue of the fundamental solution of the Poisson's equation is considered with the emphasis on the parallelization and adequacy of solution.

Введение

В нестационарных задачах астрофизики, таких как динамика волн плотности в галактиках, их спиральный узор, устойчивость колец Сатурна и других внешних планет вместе с проблемой происхождения самой Солнечной системы, необходимо находить самосогласованное гравитационное поле многих тел. Как правило, это поле зависит от всех трех пространственных переменных и определяется путем решения уравнения Пуассона для гравитационного потенциала. Зачастую для ряда указанных задач движение тел происходит в основном в одной плоскости (3D2V задачи). При этом уравнение Пуассона переходит в уравнение Лапласа с граничным условием в плоскости движения тел [1]. Для последней задачи в [2] разработаны численные методы. Они показали свою эффективность в последовательных алгоритмах при относительно небольшом (до 10^7) числе узлов расчетной сетки. Очевидным недостатком такого подхода является определение потенциала во всей трехмерной области вместо одной плоскости движения тел. Созданный параллельный алгоритм [3] оказался недостаточно эффективным в расчетах на десятках процессоров и с десятками миллионов узлов сетки, требуемых для изучения физических неустойчивостей. Кроме того, чтобы решать уравнения Пуассона и Лапласа во всей области, необходим независимый расчет граничных условий. Используемый прием определения граничных условий с необходимой для всей задачи точностью связан с удалением границ на достаточно большие расстояния, где потенциал аппроксимировался по методу моментов. Это увеличение расчетной области уменьшает число узлов и точность нахождения поля в физически важных местах. Один из выходов — расчет потенциала на границах через фундаментальное решение уравнения Пуассона. Однако этот подход ранее серьезно не рассматривался в силу того, что алгоритм требует перебора узлов сетки и вычислительно

*Работа выполнена при поддержке программ Президиума РАН “Нестационарные проблемы в астрономии” и “Происхождение и эволюция биосферы”, интеграционного проекта СО РАН (№ 148), а также Российского фонда фундаментальных исследований (грант № 02-01-00864).

© Институт вычислительных технологий Сибирского отделения Российской академии наук, 2005.

связан с очень большими затратами машинного времени [4]. Увеличение числа процессоров на суперкомпьютерах до 1000 и более, что сравнимо с размерностью сетки по одному из пространственных направлений, качественно меняет ситуацию. Целью представленной работы явилось создание и изучение параллельного алгоритма расчета граничных условий для уравнения Пуассона на основе его фундаментального решения, в частности для 3D2V задач.

1. Постановка задачи

Гравитирующие массы распределены в ограниченной плоской области с плотностью $\rho(\tau')$. Они определяют потенциал, для которого запишем в безразмерных переменных уравнение Пуассона

$$\Delta\varphi = 4\pi\rho. \quad (1)$$

Оно имеет фундаментальное решение в виде

$$\varphi(\vec{\tau}) = \int \frac{\rho(\tau')}{|\vec{R}|} d\tau', \quad (2)$$

где $|\vec{R}| = |\vec{\tau} - \vec{\tau}'|$; $\varphi(\vec{\tau})$ — это потенциал, который необходимо вычислить в точке $\vec{\tau}$. Для дискретного 2D случая в уравнении (2) заменим интеграл суммой и получим

$$\Psi_{i',j'} = \sum_{i,j} \frac{M_{i,j}}{R_{i,j,i',j'}}. \quad (3)$$

Пусть область решения определяется радиусом R_N в полярных координатах. Тогда в расчетной области введем равномерную 2D сетку потенциала $[\Psi]$ с шагами h_r и h_φ :

$$\left\{ \begin{array}{l} r_i = h_r i, \quad i = 0, 1, \dots, N_r, \quad h_r = \frac{R_N}{N_r}, \\ \varphi_j = h_\varphi j, \quad j = 0, 1, \dots, N_\varphi, \quad h_\varphi = \frac{2\pi}{N_\varphi}, \\ \Psi_{i,j} = \Psi(r_i, \varphi_j) \text{ — значение сеточной функции потенциала в узлах сетки,} \end{array} \right.$$

$\Psi_{i',j'}$ находится через суммирование вкладов точечных масс $M_{i,j}$, заданных в центрах ячеек сетки $[\Psi]$ по формуле (3). Массы составляют свою сетку $[M]$, сдвинутую относительно сетки потенциала, как показано на рис. 1. Таким образом, $R_{i,j,i',j'}$ — расстояние от узла (i', j') сетки $[\Psi]$ до узла (i, j) сетки $[M]$. Поставленная задача определения потенциала в узлах сетки на плоскости прямо связана с методом крупных частиц решения уравнения

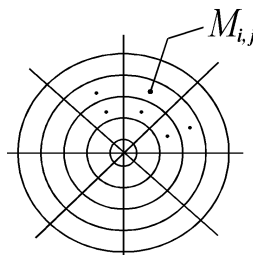


Рис. 1.

Власова — Лиувилля [1]. В задачах, включающих решение последнего уравнения, традиционно ориентируются на точность выполнения законов сохранения и расчета полей лучше 1 %. Далее будем называть методом (3) численный метод нахождения потенциала с помощью фундаментального решения уравнения Пуассона по формуле (3) на введенных сетках $[\Psi]$ и $[M]$.

2. Параллельная реализация

В программной реализации метода (3) задаются массивы Ψ потенциала и M значений точечных масс. Для распараллеливания разрежем массив масс M на np колец и/или секторов, где np — количество процессоров. В памяти каждого процессора k окажутся “его” массив масс $M_k[N(np), N_\varphi]$ и полный массив потенциала $\Psi[N_r, N_\varphi]$. В результате каждый процессор считает вклад в потенциал тех масс, которые он содержит в памяти по формуле (3). Затем массивы потенциалов из всех процессоров собираются в одном и там суммируются. Пусть $\Psi_k[N_r, N_\varphi]$ — потенциал, посчитанный в процессоре k . Тогда искомый потенциал:

$$\Psi[N_r, N_\varphi] = \sum_{k=0}^{np} \Psi_k[N_r, N_\varphi]. \quad (4)$$

Схема параллельной программы представлена на рис. 2. В блоке № 1 выполняется считывание таких входных параметров, как размерность вычислительной сетки, размер области. Эти параметры пересылаются всем процессорам, и в каждом процессоре инициализируется “его” массив M_k . После завершения работы блока № 2 полученные потенциалы пересылаются в 0-й процессор. В третьем блоке 0-й процессор готовит результат и при

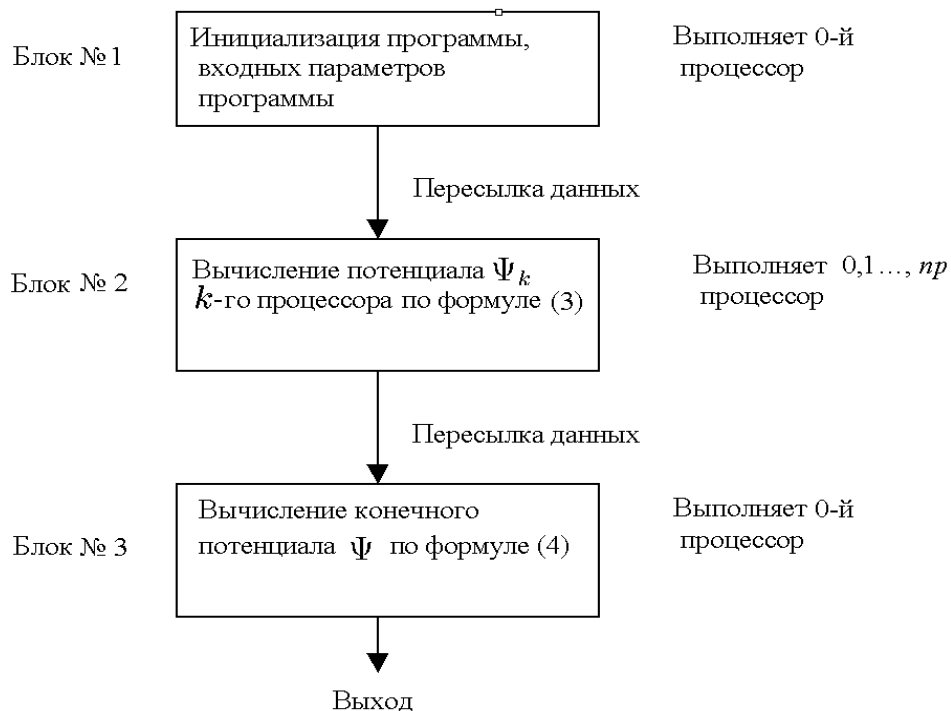


Рис. 2.

необходимости сохраняет его на диске. Обратим внимание на то, что основные затратные вычисления происходят в блоке № 2, где нет никаких пересылок данных между процессорами.

3. Сходимость и точность метода

Для исследования сходимости метода (3) использовались два теста сравнения с аналитическими решениями для бесконечно тонкого кольца и равномерного диска.

Тест с бесконечно тонким кольцом. Аналитическое решение для потенциала кольца определяется формулой

$$\Psi(r) = \frac{-2M}{\pi(r+R)} \int_0^{\pi/2} \frac{1}{\sqrt{1 - 4R \frac{r}{(r+R)^2} \sin^2(\Theta)}} d\Theta. \quad (5)$$

В расчетах кольцо имело радиус $R = 1,01$ и массу $M = 1$. Размерность сетки менялась от $(N_r = 100, N_\varphi = 100)$, $(N_r = 100, N_\varphi = 200)$, ..., до $(N_r = 100, N_\varphi = 1600)$. В табл. 1 представлены абсолютные отклонения от точного решения в области радиуса кольца, где точность наихудшая. На рис. 3 показана относительная погрешность метода в процентах для четырех сеток $N_r = 100, N_\varphi = 100$; $N_r = 100, N_\varphi = 200$; $N_r = 100, N_\varphi = 400$ и $N_r = 100, N_\varphi = 800$. Как следует из приведенных данных, есть сходимость к точному решению, и уже на третьей из приведенных сеток точность метода значительно превышает 1% во всей области. Выбор неравномерной сетки объясняется нечувствительностью кольца к дроблению сетки по радиусу.

Т а б л и ц а 1

Погрешность расчета потенциала на кольце для различных сеток

R	$N_r = 100,$ $N_\varphi = 100$	$N_r = 100,$ $N_\varphi = 200$	$N_r = 100,$ $N_\varphi = 400$	$N_r = 100,$ $N_\varphi = 800$	$N_r = 100,$ $N_\varphi = 1600$
0.94	1.48E-04	7.82E-08	2.07E-09	2.07E-09	2.07E-09
0.96	1.33E-03	5.98E-06	1.41E-10	2.41E-11	2.41E-11
0.98	1.18E-02	4.31E-04	7.40E-07	1.49E-11	1.19E-11
1.00	1.00E-01	3.14E-02	3.37E-03	4.57E-05	1.01E-08
1.02	1.01E-01	3.21E-02	3.51E-03	4.96E-05	1.19E-08
1.04	1.30E-02	5.18E-04	1.06E-06	2.11E-11	1.48E-11
1.06	1.73E-03	9.90E-06	3.28E-10	1.20E-10	1.20E-10
1.08	2.44E-04	2.16E-07	1.34E-09	1.34E-09	1.34E-09

Т а б л и ц а 2

Относительная погрешность численного решения, %

R	$N_r = 100,$ $N_\varphi = 100$	$N_r = 200,$ $N_\varphi = 200$	$N_r = 400,$ $N_\varphi = 400$	$N_r = 800,$ $N_\varphi = 800$
0.02	0.5378	0.1991	0.1036	0.0523
0.5	0.7600	0.3785	0.1895	0.0948
0.98	1.9130	0.9992	0.5022	0.2501
1	1.0691	0.5414	0.2715	0.1359
1.98	0.0001	0.0002	0.0001	0.0000

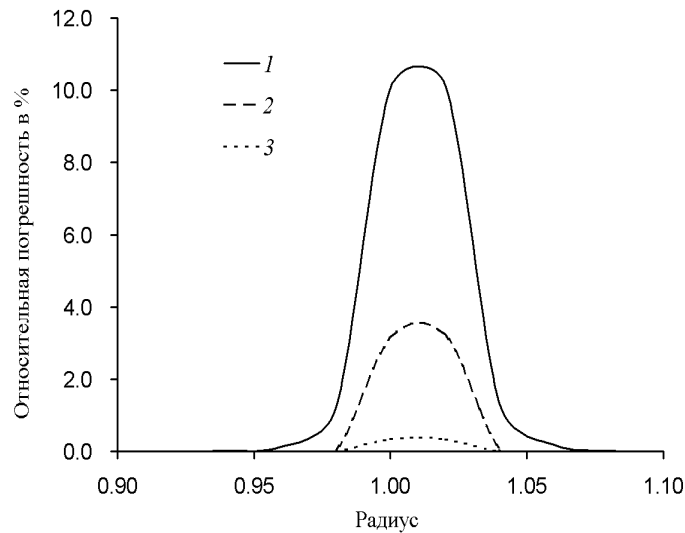


Рис. 3. Зависимость относительной погрешности от размера сетки для теста с кольцом: $N_r = 100, N_\varphi = 100$ (кривая 1); $N_r = 100, N_\varphi = 200$ (кривая 2); $N_r = 100, N_\varphi = 400$ (кривая 3).

Тест с диском. Аналитическое решение для потенциала равномерного диска определяется формулой

$$\Psi(r) = \int_0^{R_d} \frac{-4RM}{(R_d)^2 \pi (r+R)} \int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1 - 4R \frac{r}{(r+R)^2} \sin^2(\Theta)}} d\Theta dR. \quad (6)$$

В расчетах радиус диска $R_d = 1$, его масса $M = 1$. В табл. 2 представлены значения относительной погрешности, выраженной в процентах, полученного численного решения. Из данных таблицы следует примерно линейная сходимость к аналитическому решению. Наибольшее отклонение от точного значения оказалось около края диска. Эта погрешность для сетки $N_r = 200, N_\varphi = 200$ меньше 1%, для сетки $N_r = 800, N_\varphi = 800$ она составляет 0.25%.

Сравнение с другими методами. Результаты вычислений, полученных методом (3), сравнивались с решениями уравнения Пуассона комбинированным методом [2]. В частности, на тесте с равномерным диском были получены результаты расчетов обоими методами. Они сравнивались с решением (6). В данных тестах масса диска $M = 1$, радиус диска $R_d = 1$, полный радиус расчетной области $R_N = 2$. Граничные условия в методе [2] ставились на цилиндре радиуса $R = 4$, длиной 2. Размерности сеток в этих расчетах были взяты для метода (3) — $N_r = 200, N_\varphi = 200$ и для метода [2] — $N_r = 200, N_\varphi = 256$. В области от $R_N = 2$ до $R = 4$ вводилась дополнительная сетка со 100 узлами для метода [2]. В табл. 3 представлены значения относительной погрешности в процентах для комбинированного метода и для метода (3). Из данной таблицы видно, что точность рассматриваемого метода (3) в среднем не хуже, чем точность комбинированного метода [2] на более подробной сетке в этом тесте. Ухудшение точности метода [2] на границе области $r = 1.99$ связано с изменением шага сетки.

Результаты решения всей нестационарной 3D2V задачи [1] с использованием метода (3) для уравнения Лапласа на временах до нелинейных стадий развития неустойчивостей в целом оказались близкими к полученным ранее. Нелинейные стадии развития процессов

Т а б л и ц а 3
Относительная погрешность вычислений, %

r	Комбинированный метод [2]	Метод (3)
$r = 0$	0.84	0.00
$r = 0.01$	0.83	0.18
$r = 0.5$	0.71	0.38
$r = 0.99$	0.10	0.99
$r = 1$	0.37	0.54
$r = 1.99$	1.11	0.00

требуют дополнительных исследований. Последовательный алгоритм с методом (3) был медленнее комбинированного в 5–6 раз на сетках примерно $N_r = 200, N_\varphi = 256$ с двумя миллионами частиц, что вызывает интерес к параллельным реализациям метода (3).

4. Параллельная версия алгоритма

Параллельная программа для метода (3) была реализована на языке C++ с библиотекой MPI. Результаты работы параллельной версии программы полностью совпали с результатами последовательной реализации для всех проведенных тестов. Оценка эффективности параллельности была выполнена на тесте с диском, описанном выше. В табл. 4 приведены результаты тестирования параллельной программы на кластере Института катализа в монопольном режиме на сетках от ($N_r = 100, N_\varphi = 100$) до ($N_r = 800, N_\varphi = 800$). Отметим, что частота каждого процессорного элемента Intel III кластера 800 МГц. Результаты также представлены в виде графика (рис. 4) зависимости ускорения S_p от числа процессоров, где $S_p = T_1/T_p$, T_1 — время работы программы на 1-м процессоре, T_p — время работы программы на p процессорах. Из графика видно, что параллельный алгоритм метода (3) имеет высокую эффективность на данных тестах для всех представленных размерностей и при

Т а б л и ц а 4
Результаты тестирования параллельной программы

Количество процессоров	$N_r = 100, N_\varphi = 100, \text{с}$	$N_r = 200, N_\varphi = 200, \text{с}$	$N_r = 400, N_\varphi = 400, \text{мин}$	$N_r = 500, N_\varphi = 500, \text{мин}$	$N_r = 800, N_\varphi = 800, \text{мин}$
1	40	647	177	423	
2	21	376	91.4	221	1474
4	12	173	45.8	111	725
6	8	117	30.7	75	
8	6	87	22.9	56	366
10	6	70	18.3	45	
12	6	60	15.6	37	
14	6	52	13.3	33	
16	6	47	11.5	28	184
18	7	47	10.6	25	
20	8	38	9.2	22	147
22	6	36	8.7	21	
24	7	34	7.8	19	125

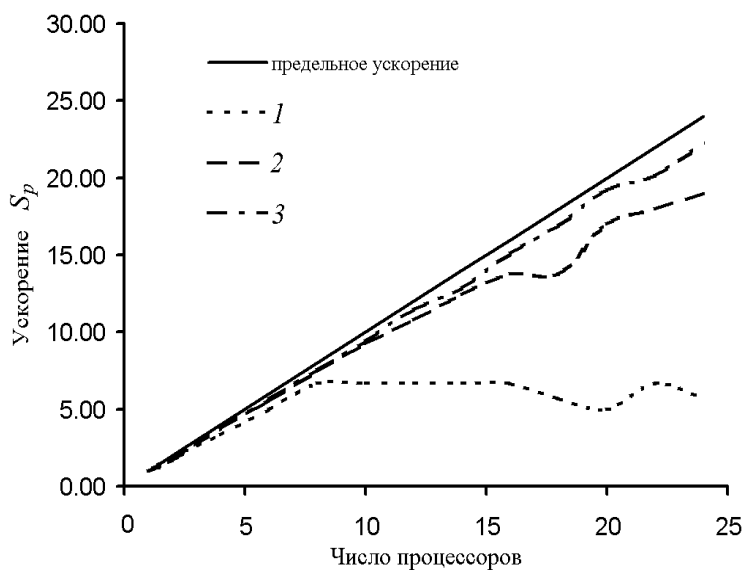


Рис. 4. Зависимость ускорения от числа процессоров для размерностей вычислительных сеток: $N_r = 100, N_\varphi = 100$ (кривая 1); $N_r = 200, N_\varphi = 200$ (кривая 2); $N_r = 500, N_\varphi = 500$ (кривая 3).

увеличении размерности вычислительной сетки эффективность метода растет. На рис. 5 приведен также график зависимости времени выполнения программы от количества процессоров, построенный по тем же данным в логарифмических координатах. Этот график полезен для прогноза необходимого количества процессоров для достижения определенной скорости выполнения программы. На обоих графиках (см. рис. 4 и 5) хорошо видно существование точки насыщения, когда увеличение числа процессоров уже не дает ускорения работы программы. Этот эффект имеется здесь только для размерности $N_r = 100, N_\varphi = 100$ на времени 6–7 с в среднем. Можно предположить, что для других размерностей эта точка также существует. Анализ показал, что время выполнения 1-го и 3-го блока схемы (см. рис. 2) пренебрежимо мало по сравнению со временем выполнения блока № 2. При этом блок № 2 не содержит межпроцессорных коммуникаций, что приводит к хорошей эффективности распараллеливания. Точку насыщения для размерности $N_r = 100, N_\varphi = 100$ можно объяснить тем, что для нее параллельный код блока № 2 выполняется настолько быстро, что время тратится в основном на последовательный код блоков № 1 и 3. Тем самым при использовании свыше 10 процессоров можно добиться ускорения вычислений в изучаемых задачах.

Обсуждение результатов. Перечислим достоинства метода, которые при использовании многих процессоров компенсируют большое число его вычислительных операций.

1. Метод прост, распараллеливается легко и естественно с коротким программным модулем, что уменьшает количество ошибок, допущенных программистом в параллельной реализации, тем самым повышает устойчивость работы программы по отношению к внешним изменениям.

2. Сопоставимая точность достигается на меньшем числе узлов сетки, кроме того, в отличие от других методов, не нужно задавать краевые условия, следовательно, можно подстроить сетку к интересующей части расчетной области.

3. В отличие от комбинированного метода, не нужна 3-я составляющая по пространству для расчета потенциала на плоскости.

4. Малое количество межпроцессорных коммуникаций приводит к хорошему ускорению

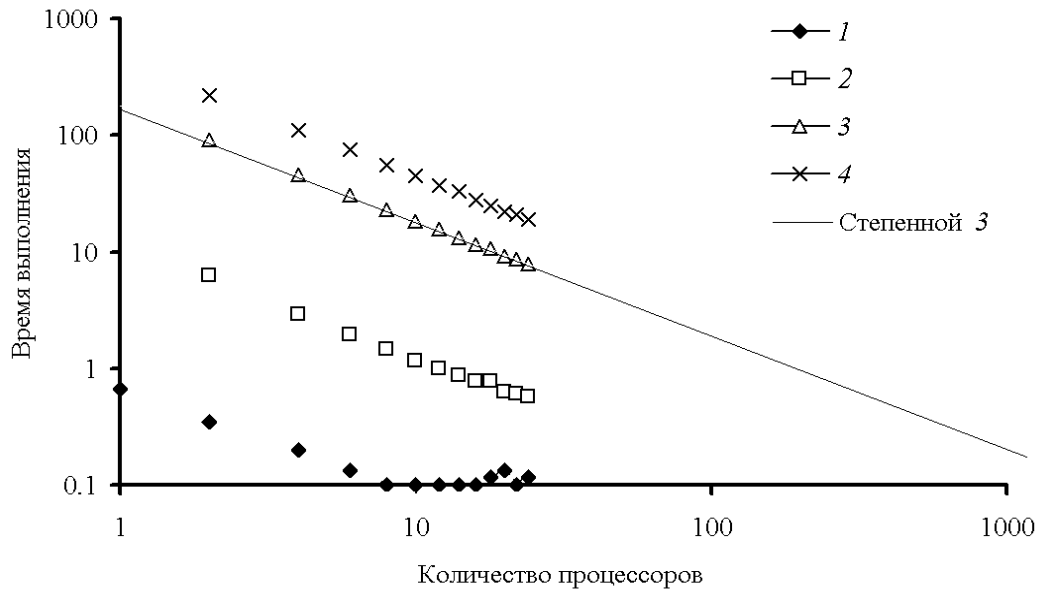


Рис. 5. Зависимость времени выполнения программы от числа процессоров в логарифмическом масштабе для размерностей вычислительных сеток: $N_r = 100, N_\varphi = 100$ (кривая 1); $N_r = 200, N_\varphi = 200$ (кривая 2); $N_r = 400, N_\varphi = 400$ (кривая 3); $N_r = 500, N_\varphi = 500$ (кривая 4).

параллельной программы при увеличении количества процессоров.

5. Метод не итерационный, поэтому можно точно прогнозировать время работы программы, что очень удобно для расчетов на удаленных суперЭВМ.

Исходя из проверенных характеристик метода (3) следует, что он может быть применен для нахождения потенциала на плоскости в задаче 3D2V на многопроцессорных ЭВМ. По сравнению с комбинированным методом [2] метод экономит оперативную память, что позволяет увеличивать расчетную сетку, а на большом (100 и более) количестве процессоров можно получить решение недоступных ранее по размерностям сетки задач. Для 3D3V задач метод может быть применен в расчете граничных условий. Эти граничные условия интересны и для создания эффективных параллельных алгоритмов.

Заключение

В работе исследован дискретный аналог фундаментального решения уравнения Пуассона с точки зрения эффективности его распараллеливания, а также точности по отношению к аналитическим решениям. Показано, что метод может быть востребован для решения уравнения Пуассона при использовании суперЭВМ с большим (свыше 20) числом процессоров в первую очередь для расчета граничных условий в полностью трехмерных задачах. В частности, применение этого метода оправданно для решения 3D2V задач. Метод идеально подходит для работы именно на параллельных системах. Простота метода позволяет резко сократить время его реализации на суперЭВМ в сравнении с другими методами решения уравнения Пуассона.

Авторы выражают благодарность В.А. Вшивкову за критические замечания и полезные консультации.

Список литературы

- [1] Снытников В.Н., Вшивков В.А., Кукшева Э.А. и др. Трехмерное численное моделирование нестационарной гравитирующей системы многих тел с газом // Письма в астрономический журнал. 2004. Т. 30, № 2. С. 146–160.
- [2] Снытников В.Н., Пармон В.Н., Вшивков В.А. и др. Численное моделирование гравитационных систем многих тел с газом // Вычисл. технологии. 2002. Т. 7, № 3. С. 72–84.
- [3] Вшивков В.А., Кукшева Э.А., Никитин С.А. и др. О параллельной реализации численной модели физики гравитирующих систем // Автометрия. 2003. Т. 39, № 3. С. 115–123.
- [4] Олдер Б., Фернбах С., Ротенберг М. Вычислительные методы в физике плазмы. М.: Мир, 1974.

*Поступила в редакцию 9 ноября 2004 г.,
в переработанном виде — 21 декабря 2004 г.*