УДК 519.6

МЕХАНИЗМ АДАПТИВНОГО ВЫБОРА РЕШАТЕЛЯ В БИБЛИОТЕКЕ PMLP/PARSOL

В. А. Ерзунов, А. А. Горбунов
(РФЯЦ-ВНИИЭФ)

Дается описание механизма адаптивного выбора решателя в библиотеке PMLP/Parsol и приведены результаты численных экспериментов при расчетах в параллельном режиме по методике МЕДУЗА, предназначенной для расчета задач математической физики на нерегулярных сетках. Механизм адаптивного выбора решателя позволяет в процессе счета автоматически подобрать оптимальный решатель для всех этапов расчета задачи, что экономит время счета.

Ключевые слова: библиотека решателей, предобусловливатель, методика МЕДУЗА, решение уравнения теплопроводности.

Введение

В ряде программных комплексов, применяемых в РФЯЦ-ВНИИЭФ для решения задач математической физики, используется библиотека PMLP/Parsol [1], предназначенная для решения итерационными методами разреженных линейных систем (СЛАУ) в скалярном и параллельном режимах. В состав библиотеки входит большое число решателей,* разработанных для решения линейных систем различной степени сложности. Большинство задач, для расчета которых используется библиотека PMLP/Parsol, представляют собой развивающиеся во времени процессы, при этом сложность линейных систем с течением времени может меняться. В программных комплексах, использующих библиотеку, решатель обычно задается один раз в начале счета задачи, что явно невыгодно с точки зрения экономии машинного времени.

В настоящее время в библиотеке PMLP/Parsol разработан и реализован механизм адаптивного выбора решателя, позволяющий в процессе счета автоматически подобрать оптимальный решатель для любого этапа расчета задачи. Механизм предназначен для использования в процессах, имеющих циклический характер, таких как итерационный процесс при решении уравнения теплопроводности с помощью неявной разностной схемы. Тогда цикл представляет собой один шаг по времени расчета уравнения теплопроводности. Идея заключается в построении шкалы сложности решателей на первом цикле и выборе решателя на основании построенной шкалы на последующих циклах с постоянным контролем степени сложности линейных систем. При существенном изменении сложности систем шкала автоматически перестраивается.

Методика МЕДУЗА [2] предназначена для расчета задач математической физики на нерегулярных сетках в скалярном и параллельном режимах. В модуле расчета процесса теплопроводности используется библиотека линейных решателей PMLP/Parsol. Матрицы ленточного типа, формирующиеся в процессе счета по МЕДУЗЕ, отличаются большой шириной ленты и отсутствием ярко выраженной блочной структуры, а также значительной сложностью. При этом в процессе счета их структура и степень сложности могут существенно меняться. Поэтому при расчетах по методике МЕДУЗА в качестве предобусловливателя обычно задается один из самых сложных (предобусловливатель типа LU с высокой точностью разложения), в качестве решателя — один из решателей семейства BiCgStab, что обеспечивает гарантированное решение систем практически любого вида. Обес-

*Здесь и далее, за исключением особо оговариваемых случаев, под решателем будем понимать связку: *предобусловливатель + итерационный решатель*.

печивая решение практически любой системы на всех стадиях расчета, естественно, решатель не является оптимальным с точки зрения скорости счета. Механизм адаптивного выбора решателя был включен в модуль расчета процесса теплопроводности методики МЕДУЗА [3].

В работе описан механизм адаптивного выбора решателя и приведены результаты численных экспериментов при расчетах в параллельном режиме по методике МЕДУЗА.

Механизм адаптивного выбора решателя

Идея создания механизма адаптивного выбора решателя заключается в выборе непосредственно в процессе расчета наиболее подходящего решателя для задачи в каждый конкретный момент времени средствами библиотеки линейных решателей на основе построенной шкалы сложности решателей.

Введем понятия *сложности решателя* и *сложности линейной системы.*

Решатель будем считать *простым*, если он строится быстро, и *сложным* в противном случае.

Сложность любой линейной системы определяется временем ее решения, т. е. сложность системы можно измерить числом итераций, требуемых для ее решения с заданной точностью каким-либо решателем. Другая составляющая, входящая во время решения, а именно время построения решателя, является практически постоянной величиной для конкретного решателя при условии незначительного изменения размеров линейной системы и ее структуры. Поэтому критерием для смены решателя может служить число итераций, необходимых для решения линейной системы. Если существует шкала, в которой можно задать минимальное и максимальное число итераций для каждого решателя, то выход за эти пределы служит критерием смены решателя. Способы создания такой шкалы (*ScalePreconds*) будут описаны в следующем разделе.

Процесс решения системы начинается с решателя, который был выбран как наиболее подходящий. Смена решателя в ходе решения, например в программе расчета уравнения теплопроводности, основана на предположении плавности изменения сложности решаемых линейных систем. После решения очередной линейной системы происходит анализ числа затраченных на ее решение итераций и кода завершения решения.

Если код завершения сигнализирует о том, что за заданное число итераций (из шкалы *ScalePreconds*) требуемая точность не достигнута, строится новый, более сложный решатель и решение повторяется с уже полученным ранее начальным приближением, число итераций при этом запоминается. Выход из процедуры решения произойдет только после получения решения с заданной точностью по заданному критерию сходимости. Так происходит движение в сторону выбора более сложного решателя из шкалы. Максимальное число итераций для самого сложного решателя — один из параметров, задаваемых пользователем. И только в случае невозможности нахождения решения последним (самым сложным) решателем с заданной точностью за максимально допустимое число итераций об этом соответствующим кодом сообщается пользователю.

Если после решения очередной линейной системы заданная точность достигнута, то анализируется число итераций. Если оно меньше минимального для данного решателя (из шкалы *ScalePreconds*), осуществляется подготовка построения для решения следующей линейной системы, менее затратной по времени построения, т. е. более простой, и происходит выход из процедуры решения. Так идет движение в сторону упрощения решателя из шкалы.

В настоящее время реализована возможность выбора наиболее подходящего решателя двумя способами. При первом способе выбор решателя осуществляется именно так, как описано выше, без учета предыдущей истории решений. Второй, более сложный способ выбора решателя, предназначен для тех процессов, в которых изменение сложности решения линейных систем периодически (циклически) повторяется. К ним относятся задачи расчета уравнения теплопроводности, для которых обычно строится неявная разностная схема, решаемая одним из итерационных методов.

Периодическое изменение сложности линейных систем происходит в ходе итерационного процесса по нелинейности уравнения состояния и коэффициента теплопроводности, при этом периодом обычно является один шаг по времени (на каждой итерации по нелинейности уравнения состояния и коэффициента теплопроводности решается одна линейная система). В этом случае при первом прохождении решений линейных систем на участке от начала до конца итераций по нелинейности уравнения состояния и коэффициента теплопроводности строится цепочка истории решений, в которой для каждого решателя отмечается дополнительно количество успешно решенных линейных систем. Поэтому при очередном успешном решении линейной системы в случае использования истории решений, если число итераций попадает в отведенные пределы, число уже решенных линейных систем сравнивается с тем, что занесено в историю на предыдущем цикле решений. Если число текущих успешных решений достигло предела, то, несмотря на то, что число полученных итераций находится в разрешенном диапазоне, для последующего построения и решения выбирается следующий решатель из истории. Если же решение закончилось неудачей, т. е. заданная точность не достигнута или число итераций вышло за отведенные пределы, то уже сформированная на предыдущем шаге цикла история забывается и достраивается заново. История решений записывается в файл по концу задачи, для того чтобы при продолжении счета его можно было сразу же использовать.

Ниже приведен пример файла истории решений, полученного при расчете одной из задач по методике МЕДУЗА:

```
#num_HistoryPreconds
4
#index_ScalePreconds num_solves
4                    1
3                    3
2                    1
1                    1
```

Здесь для расчета использовались 4 решателя. В первом столбце приведен номер решателя, во втором — число систем, решенных данным решателем.

Историю решений, записанную в файле, можно прокомментировать следующим образом. В истории решений было сделано 4 записи, сначала решалась одна линейная система с решателем под номером 4 из шкалы *ScalePreconds*, затем три линейные системы с решателем под номером 3, затем по одной соответственно с решателями под номерами 2 и 1. Отметим, что при нумерации решателей степень сложности решателя убывает с уменьшением его номера N_r.

Рис. 1 иллюстрирует файл истории решений. На рисунке изображен характер изменения сложности линейных систем в процессе одного шага по времени решения уравнения теплопроводности (номер линейной системы N_s есть номер итерации по нелинейности уравнения состояния и коэффициента теплопроводности). Сложность линейных систем в задаче постепенно понижалась к концу цикла.

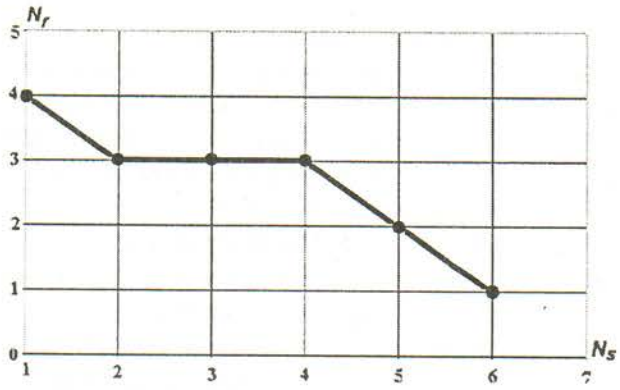

Рис. 1. Изменение характера сложности линейных систем в процессе одного шага по времени решения уравнения теплопроводности

Построение шкалы сложности решателей

Прежде чем описывать способы получения шкалы решателей, приведем файл шкалы решателей *ScalePreconds.pmlp*, который получен при расчете одной из задач по методике МЕДУЗА:

```
# size of solvers & number of begin
# solvers
6 3
# numbers of main solvers
259 297 297 297 297 297
# numbers of inner solvers
7 7 7 7 7 7
# parameter for main & inner solvers
0.0 0.01 0.001 0.0001 0.00001 0.000001
# numbers of iters in
# DDomanDecomposition
1 1 1 1 1 1
# min iters numbers for go to prev.
0 2 2 4 3 2
# max iters numbers for go to next.
2 2 6 11 7 200
```

В файле задаются число решателей, наиболее подходящий номер решателя для начала решения и наборы, описывающие параметры для построения решателей разной степени сложности. В каждый такой набор входит номер решателя и параметры для его построения. Даже если номера решателей совпадают, но отличаются хотя бы одним из параметров, например степенью разложения для предобусловливателей типа LU-разложения, то это будут разные по сложности и по применимости решатели.

Конечной целью при построении шкалы решателей является вычисление для каждого решателя допустимого диапазона итераций, при котором он имеет право быть использованным при решении (последние четыре строки приведенного файла, включая комментарии).

Исходным материалом для построения шкалы решателей служит набор решателей с параметрами, поставляемый библиотекой.

Существует три способа получения шкалы решателей.

Первый способ — задать все параметры вручную, исходя из знаний о решателях и особенностях методики. Недостатки очевидны, достоинства заключаются в том, что не требуется машинного времени.

Второй способ — построение файла со шкалой решателей по примерам характерных матриц и векторов правых частей линейных систем, предоставленных пользователями. Была разработана и реализована самостоятельная программа, которая считывает матрицы и векторы правых частей и по некоторым алгоритмам строит шкалу решателей. Достоинством является простота способа построения шкалы. К недостаткам этого способа следует отнести необходимость иметь наборы линейных систем для всего класса рассчитываемых каждой конкретной методикой задач и постоянно их обновлять, а также необходимость отдельного запуска программы построения шкалы решателей в тех же условиях и с той же точностью, что и в методике, в которой предполагается использование построенной шкалы.

Третий способ — построение шкалы решателей непосредственно в методике. Основным достоинством данного подхода является возможность быстрого и точного построения шкалы для конкретной задачи. Недостатками являются некоторое удорожание решения на первом цикле и то, что на начальном этапе расчета линейные системы могут оказаться нехарактерными для дальнейших этапов счета, т. е. могут происходить перестройки шкалы решателей.

Поскольку второй и третий способы основаны на обращении к одним и тем же программам, более подробно остановимся на описании последнего способа построения.

Для начала из списка кандидатов для использования в шкале решателей убираются те решатели, которые не могут быть применены в конкретных условиях. Например, для решения линейной системы, полученной из трехмерной сетки, не годятся решатели, рассчитанные только на двумерную решетку; для скалярной задачи отбрасываются блочные решатели с двумя и более внешними итерациями в предобусловливателе и т. п. Далее решается некоторое число линейных систем, причем каждая решается всеми оставшимися решателями. При этом запоминается время построения, время решения, число итераций, необходимых для получения решения с заданной точностью по заданному критерию для каждого решателя. Эти значения усредняются для определения *цены* одной итерации для каждого решателя. Затем из списка кандидатов убираются те, которые не справились с решением всех предложенных линейных систем за максимально допустимое число итераций. Список кандидатов сортируется по возрастанию времени построения (хотя изначально были варианты сортировки по убыванию времени решения и построения) и определяется наилучший кандидат для начала решения. От него с учетом коэффициента понижения или повышения сложности линейной системы вычисляются границы перехода на соседние решатели, т. е. минимальное и максимальное число итераций.

В этом алгоритме присутствует дополнительный параметр — коэффициент повышения или понижения сложности линейной системы, с помощью которого можно изменять плавность переходов между решателями. Приведенный алгоритм — достаточно затратный по времени, но он выполняется только на начальном этапе (на первом цикле) и в дальнейшем его доля в общем времени решения почти не ощущается.

Затем надо определить количество линейных систем, которые необходимо решить при работе в таком режиме. Для второго способа построения этот вопрос отсутствует. А вот для построения шкалы решателя непосредственно из методики количество систем задается в зависимости от режима дальнейшей работы — с запоминанием истории решений или без нее. В первом случае для решения используются все линейные системы на первом цикле, во втором случае задается конкретное количество решаемых линейных систем в исходных параметрах при обращении к библиотеке.

Отметим следующий момент. Исследования, проведенные на задачах, рассчитанных по методике МЕДУЗА, показали, что изменение сложности линейных систем имеет только один скачок между циклами — обычно это переход от последней линейной системы предыдущего шага по времени к первой линейной системе текущего шага. Это связано с тем, что в задачах обычно рассчитываются процессы газовой динамики и теплопроводности, причем на этапе расчета процесса газовой динамики при переходе с одного шага по времени на другой происходит существенное изменение термодинамических величин. В начале каждого цикла выбираются наиболее сложные для решения линейные системы, которые в процессе цикла (в процессе итераций по нелинейности уравнения состояния и коэффициента теплопроводности на шаге по времени) становятся все проще и проще. Для таких задач лучше всего подходит способ выбора решателя с запоминанием истории с предыдущего цикла.

Результаты численных исследований

Описанный механизм адаптивного выбора решателя был внедрен в модуль расчета уравнения теплопроводности методики МЕДУЗА. Исследования проводились на четырех задачах:

1. Двумерная задача, в которой происходит газодинамическое сжатие и прогрев сферической системы веществ. Задача считалась в 4-, 8-, 16-процессорном режимах.
2. Двумерная задача, в которой тепло распространяется по изначально холодной системе веществ, далее происходит газодинамическое сжатие части веществ.
3. Двумерная задача, геометрия задачи — сфера.
4. Задача характеризуется существенным изменением сложности решения линейных систем в процессе расчета.

Задачи 1, 2, 4 имеют размер сетки около 140 000 ячеек, задача 3 — около 1 000 000 ячеек.

Расчеты проводились в параллельном режиме на 16 и 56 процессорах. Все расчеты выполнялись в режиме с построением шкалы решателей и с хранением истории решений. На каждом шаге по времени запоминались время выполнения всей задачи и время решения всех линейных систем. Результаты расчетов с применением механизма адаптивного выбора решателя сравнивались с результатами расчетов в режиме с заданием конкретного решателя (штатном режиме), с использованием предобусловливателя типа LU с высокой точностью разложения.

На рис. 2—11 и в табл. 1—5 показаны результаты работы механизма адаптивного выбора решателя на этих задачах.

Обозначения на рисунках: t_h — время счета шага (с); N_h — число шагов по времени; t_s — время решения линейной системы; N_r — номер решателя; N_s — номер линейной системы.

По результатам расчетов можно сделать следующие выводы:

1. Время решения линейной системы сократилось в 1,18—19,9 раза по сравнению с временем расчета в штатном для методики режиме.

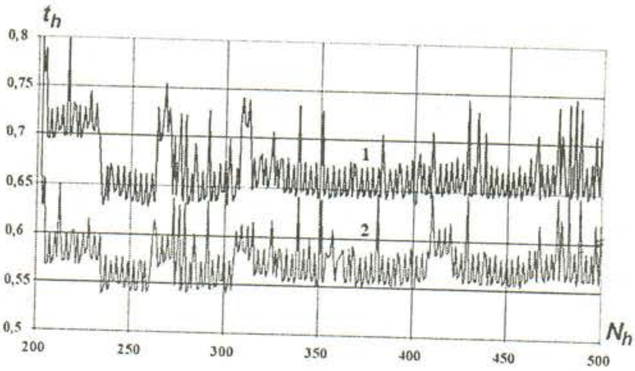

Рис. 2. Задача 1. График зависимости полного времени счета одного временного шага от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

Рис. 3. Задача 1. График зависимости времени решения линейной системы на одном временном шаге от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

2. Общее время счета в параллельном режиме сократилось в 1,07—2,08 раза по сравнению с временем расчета в штатном режиме.

3. Доля времени на решение линейной системы сократилась с 10,9—58,8 % до 2,6—7,7 % от общего времени счета задачи.

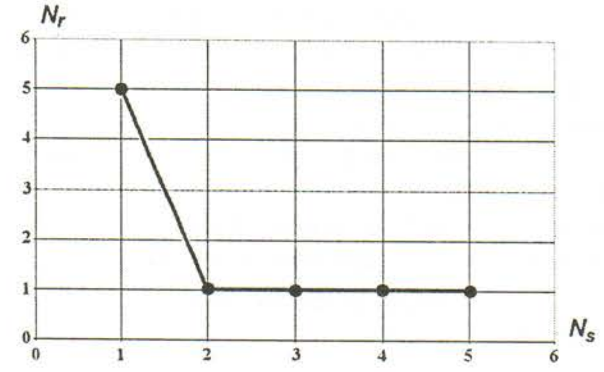

Рис. 4. Задача 1. Динамика изменения сложности линейных систем в процессе одного из временных шагов решения уравнения теплопроводности

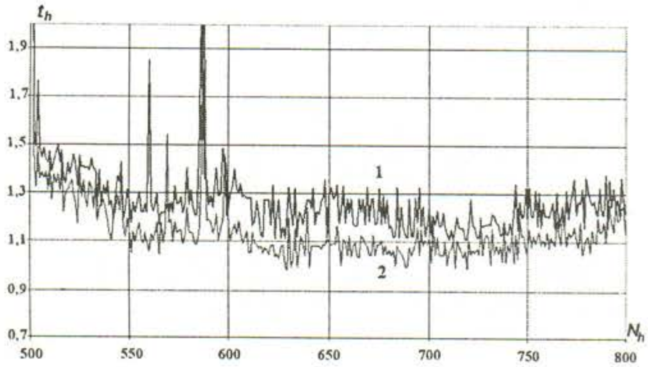

Рис. 5. Задача 2. График зависимости полного времени счета одного временного шага от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

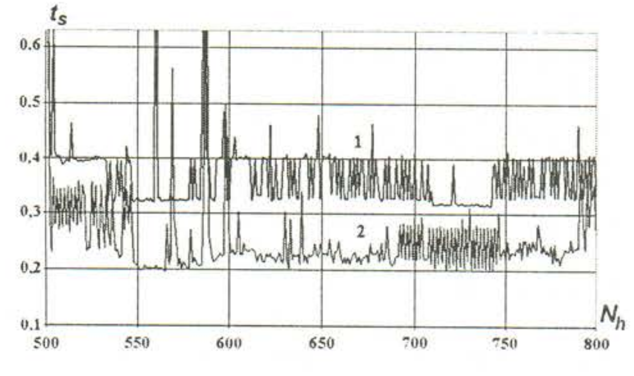

Рис. 6. Задача 2. График зависимости времени решения линейной системы на одном временном шаге от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

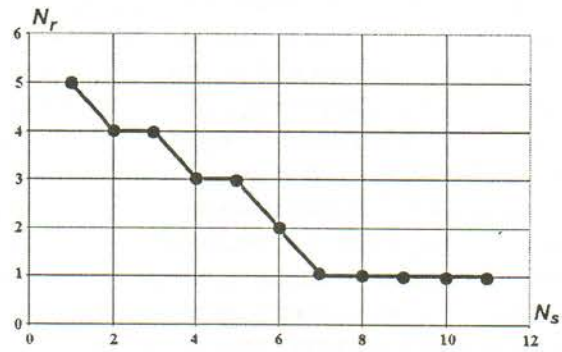

Рис. 7. Задача 2. Изменение сложности линейных систем в процессе одного из временных шагов решения уравнения теплопроводности

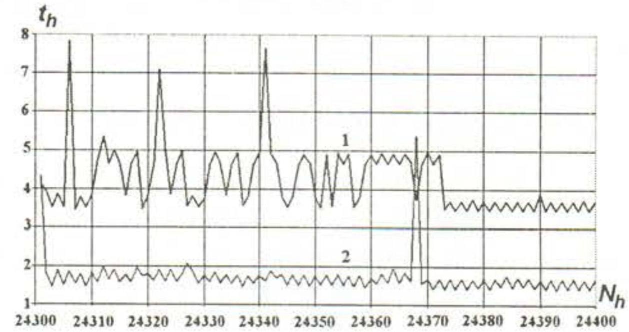

Рис. 8. Задача 3. График зависимости полного времени счета одного временного шага от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

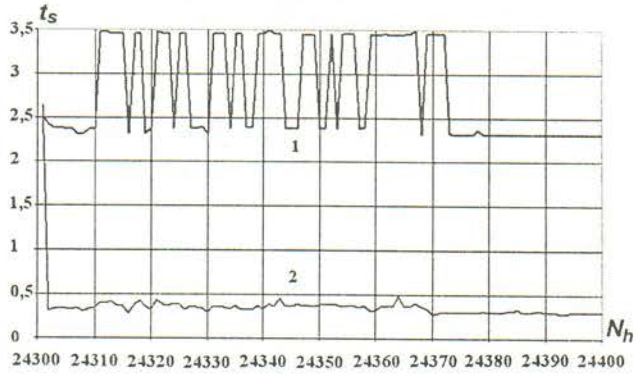

Рис. 9. Задача 3. График зависимости времени решения линейной системы на одном временном шаге от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

Таблица 1

Времена работы решателей при расчетах задачи 1 с использованием различных режимов счета. Число процессов — 16, выполнено 500 шагов по времени

Режим	Время работы решателя, с			Общее время счета задачи, с	Доля решателя в общем времени счета задачи, %
	Построение решателя	Решение системы	Суммарное время (решение + построение)		
Без использования механизма адаптивного выбора решателя	31,3	4,9	36,2	333,1	10,9
С использованием механизма адаптивного выбора решателя	2,2	5,5	7,7	301,2	2,6
Коэффициент ускорения	14,2	0,89	4,7	1,11	—

Таблица 2

Времена работы решателей при счете задачи 1 в параллельном режиме на различном числе процессоров. Выполнено 500 шагов по времени

Число процессов	Без адаптивного выбора решателя		С адаптивным выбором решателя		Ускорение
	Время решателя, с	Эффективность распараллеливания, %	Время решателя, с	Эффективность распараллеливания, %	
4	76,2	—	26,1	—	2,9
8	42,2	90,3	13,0	100	3,25
16	36,2	52,6	7,7	84,7	4,7

Таблица 3

Времена работы решателей при расчетах задачи 2 с использованием различных режимов счета. Число процессов — 16, выполнено 800 шагов по времени

Режим	Время работы решателя, с			Общее время счета задачи, с	Доля решателя в общем времени счета задачи, %
	Построение решателя	Решение системы	Суммарное время (решение + построение)		
Без использования механизма адаптивного выбора решателя	81,2	17,05	98,25	541,5	18,14
С использованием механизма адаптивного выбора решателя	49,9	17,6	67,5	440,7	15,3
Коэффициент ускорения	1,63	0,97	1,46	1,23	—

Таблица 4

Времена работы решателей при расчетах задачи 3 с использованием различных режимов счета. Число процессов — 56, выполнено 100 шагов по времени

Режим	Время работы решателя, с			Общее время счета задачи, с	Доля решателя от общего времени счета задачи, %	Эффективность распараллеливания, %
	Построение решателя	Решение системы	Суммарное время (решение + построение)			
Без использования механизма адаптивного выбора решателя	248,8	12,08	260,88	443,9	58,8	37
С использованием механизма адаптивного выбора решателя	7,2	9,2	16,4	213,3	7,7	67
Коэффициент ускорения	34,5	1,3	19,9	2,08	—	—

Таблица 5

Времена работы решателей при расчетах задачи 4 с использованием различных режимов счета. Число процессов — 16, выполнено 599 шагов по времени

Режим	Время работы решателя, с			Общее время счета задачи, с	Доля решателя в общем времени счета задачи, %
	Построение решателя	Решение системы	Суммарное время (решение + построение)		
Без использования механизма адаптивного выбора решателя	50,08	6,58	56,66	527,7	10,7
С использованием механизма адаптивного выбора решателя	4,2	6,03	10,23	445,7	2,3
Коэффициент ускорения	11,9	1,09	5,54	1,18	—

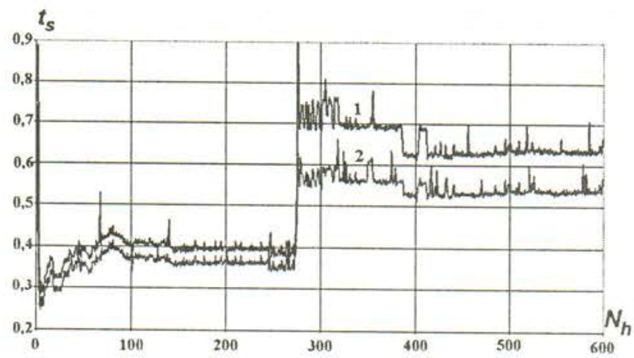

Рис. 10. Задача 4. График зависимости полного времени счета одного временного шага от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

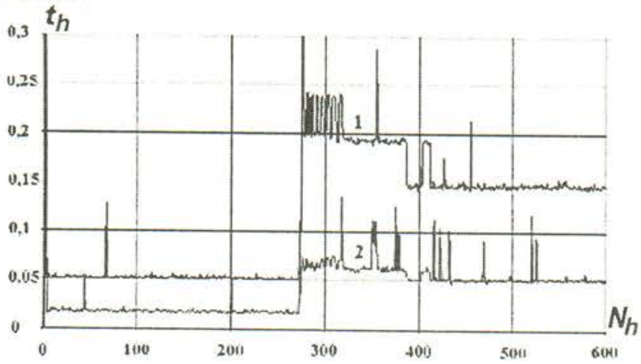

Рис. 11. Задача 4. График зависимости времени решения линейной системы на одном временном шаге от номера шага: 1 — в штатном режиме; 2 — с адаптивным выбором решателя

Заключение

В библиотеке линейных решателей PMLP/Parsol разработан и реализован механизм адаптивного выбора решателя. Это не создание новых решателей и предобусловливателей — это более правильное использование уже имевшихся возможностей библиотеки. Пользователь освобожден от задачи выбора оптимального решателя.

Время решения линейной системы и общее время счета в параллельном режиме по методике МЕДУЗА существенно сократилось по сравнению со временем расчета в штатном режиме.

Список литературы

1. *Артемьев А. Ю., Бартенев Ю. Г., Басалов В. Г. и др.* Библиотека решателей разреженных линейных систем // Труды РФЯЦ-ВНИИЭФ. 2004. Вып. 7. С. 80—95.

2. *Глаголева Ю. П., Жогов Б. М., Кирьянов Ю. Ф. и др.* Основы методики МЕДУЗА численного расчета двумерных нестационарных задач газодинамики // Числ. методы мех. спл. среды. 1972. Т. 3, № 2. С. 18—55.

3. *Горбунов А. А.* Метод решения уравнения теплопроводности на нерегулярных сетках в параллельном режиме в методике МЕДУЗА // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2008. Вып. 3. С. 32—45.

Статья поступила в редакцию 25.08.08.