

Исследование эффективности применения технологии DVM/DVMH для параллельного решения одномерных параметрических задач газовой динамики

Бондарев А.Е., Кувшинников А.Е., ИПМ им. М.В. Келдыша РАН
bond@keldysh.ru, kuvch90@yandex.ru

Аннотация

Данная работа представляет собой исследование эффективности применения технологии DVM/DVMH для параллельного решения одномерных параметрических задач газовой динамики. Приведено описание проведенных численных экспериментов по исследованию вопроса нахождения оптимального соотношения между задаваемым числом вычислительных узлов, задаваемыми параметрами этих узлов и выделяемых графических сопроцессоров для различных задач на конкретных программных кодах, представляющих одномерные параметрические задачи газовой динамики. Представлены результаты расчетов и характеристики эффективности и ускорения.

1 Введение

Параллельные вычисления являются одной из определяющих черт современного мира. Тенденция к росту применения параллельных вычислений в самых разнообразных отраслях человеческой деятельности развивается неуклонно и неустанно. Постоянно возникают новые, все более мощные и эффективные многопроцессорные вычислительные системы, появляются новые архитектуры высокопроизводительных вычислительных систем. Это сопровождается интенсивным ростом числа процессоров, используемых для вычислений в подобных системах. Интенсивное современное развитие новых гетерогенных и гибридных компьютерных архитектур, в частности, на основе многоядерных вычислительных ускорителей, позволяет значительно повысить производительность суперкомпьютеров.

Одним из важнейших и по сей день не оцененных в достаточной степени преимуществ параллельных вычислений является то, что они позволяют параллельно решать в многозадачном режиме одну и ту же зада-

чу с разными входными данными. Подобное преимущество делает параллельные вычисления важнейшим инструментом для решения оптимизационных задач, обратных задач в оптимизационной постановке и задач параметрического поиска. Эти задачи выходят на первый план в современных исследованиях. Характерной общей чертой этих типов задач является то, что в вычислительном смысле они сводятся к массовому решению однотипных задач с изменяющимися входными параметрами. Это обстоятельство делает их удобными объектами для параллельных вычислений на основе многозадачного параллелизма. При решении подобных задач результаты представляют собой большие многомерные объемы данных, анализ и обработка которых, в свою очередь, также проводится с помощью параллельных вычислений. Без применения параллельных вычислений решение задач подобного типа не представляется возможным. Особенности задач подобного типа и применение параллельных вычислений к данным типам задач подробно описываются в работах [1-6].

Разработанная в Институте прикладной математики им. М.В. Келдыша РАН высокоуровневая модель параллельного программирования DVMH (DVM for Heterogeneous systems) [7] предназначена для разработки параллельных программ численного моделирования на языках C-DVMH и Fortran-DVMH. Эти языки используют единую модель параллельного программирования (DVMH-модель) и являются расширением стандартных языков Си и Фортран спецификациями параллелизма, оформленными в виде директив компилятору. Поскольку директивы невидимы для стандартных компиляторов, программист может иметь одну программу и для последовательного, и для параллельного выполнения на ЭВМ разной архитектуры.

DVMH-модель [8-10] позволяет создавать эффективные параллельные программы (DVMH-программы) для гетерогенных вычислительных кластеров, в узлах которых в качестве вычислительных устройств наряду с универсальными многоядерными процессорами могут использоваться ускорители (графические процессоры или сопроцессоры Intel Xeon Phi). При этом отобранные на узел вычисления могут автоматически распределяться между вычислительными устройствами узла с учетом их производительности. Компиляторы языков C-DVMH и Fortran-DVMH преобразуют входную программу в параллельную программу, использующую стандартные технологии программирования MPI, OpenMP и CUDA.

В состав DVM-системы входят средства функциональной отладки и отладки эффективности DVMH-программ. Наличие такого эффективного инструмента для распараллеливания программ позволяет широчайший спектр задач, в том числе решать оптимизационные задачи, обратные задачи в оптимизационной постановке и задачи параметрического поиска. Результатом решения таких задач являются многомерные массивы информации, подлежащие отдельному анализу и обработке.

Продолжая исследования работ [1-6], данная работа имеет своей целью описание результатов проведенных вычислительных экспериментов в рамках разработки исследовательского прототипа программной системы параллельного решения оптимизационной задачи и задачи параметрического поиска в информационном массиве данных с использованием параллельных технологий DVMH на гибридном вычислительном кластере.

2 Описание тестовых задач и результаты расчетов

Для исследований эффективности применения системы DVM/DVMH был выбран ряд тестовых задач. Все эти задачи рассматривались как параметрические, то есть, для некоторых параметров задачи в заданных диапазонах задавалось разбиение, и затем задачи решались в параллельном многозадачном режиме с помощью системы DVMH. Все выбранные задачи являются простыми и достаточно известными одно-

мерными задачами вычислительной газовой динамики. Рассмотрим эти задачи вместе с алгоритмами решения и полученные результаты.

Линейное уравнение Бюргерса

Рассмотрим краевую задачу, основанную на применении линейного уравнения Бюргерса, представленного в следующем виде:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = \varepsilon \frac{\partial^2 u}{\partial x^2} + f(x, t)$$

Начальные и граничные условия представляются в виде:

$$u(x, t_0) = y(x, t_0), \quad u(0, t) = y(0, t), \\ u(L, t) = y(L, t)$$

Где $y(x, t)$ – точное решение, описывающее функцию типа бегущей волны.

Точное решение $y(x, t)$ имеет вид:

$$y(x, t) = y(\xi) = e^{-d(1-2\xi)^2}, \quad \text{где } \xi = \frac{t-x}{t_1}$$

Величины t_1 и d характеризуют ширину и крутизну волны и являются параметрами задачи.

Функция $f(x, t)$ в правой части является выражением вида:

$$f(x, t) = -\frac{\varepsilon}{t_1^2} [16d^2(1-2\xi)^2 - 8d] e^{-d(1-2\xi)^2}$$

Для решения данной задачи использовалась неявная конечно-разностная схема, подробно описанная в [6]. При решении параметрической задачи параметры ε (коэффициент вязкости) и коэффициент разностной схемы s_k разбивались в определенных диапазонах, и для каждой пары значений решалась описанная выше задача. При проведении экспериментов по распараллеливанию данного программного кода с помощью DVM варьировались следующие параметры:

N — число MPI процессов,

PPN — число MPI процессов на один вычислительный узел.

В проведенных экспериментах число MPI процессов N варьировалось от 1 до 32, а число MPI процессов на один вычислительный узел PPN варьировалось от 1 до 8. Результаты представлены ниже в таблице 1.

Здесь и далее в таблицах будем использовать следующие обозначения:

N — число MPI-процессов,

PPN — число MPI-процессов, запускаемых на одном узле,

THR — число нитей, используемых каждым MPI-процессом,

CUDA — число графических ускорителей, используемых каждым MPI-процессом,

T — время в секундах, S — ускорение T/T_{serial} ,

E — эффективность параллелизации.

Табл. 1. Уравнение Бюргерса-неявная схема

N	PPN	T	S	E
serial	-----	51,4	1	1
1	1	51,4	1,001	1,001
2	1	25,7	1,997	0,998
2	2	25,7	1,998	0,999
4	1	12,9	3,988	0,997
4	2	12,9	3,987	0,997
4	4	12,9	3,991	0,998
8	1	7,61	6,753	0,844
8	2	6,71	7,662	0,958
8	4	6,73	7,641	0,955
8	8	6,98	7,363	0,920
16	2	4,09	12,580	0,786
16	4	3,64	14,129	0,883
16	8	3,76	13,652	0,853
32	2	2,08	24,684	0,769
32	4	2,09	24,613	0,769
32	8	2,16	23,782	0,743

Полученные результаты расчетов показали эффективность реализованного решения по многозадачному распараллеливанию. Одновременно в работе [6] отмечалось, что применяемая в программном коде решения уравнения Бюргерса неявная конечно-разностная схема (WW-схема) неудобна для распараллеливания с помощью подключения графических сопроцессоров. Для проведения следующей серии численных тестов ее было необходимо заменить на явный аналог с сохранением свойств аппроксимации и устойчивости. Для проведения дальнейших экспериментов по подключе-

нию графических сопроцессоров была проведена замена используемой в алгоритме конечно-разностной схемы. Неявная конечно-разностная схема [6] (WW-схема) была заменена в алгоритме и программном коде на явную схему типа Лакса-Вендроффа, имеющую второй порядок по пространству и времени:

$$u_i^{n+1} = u_i^n - \frac{\tau}{2h}(u_{i+1}^n - u_{i-1}^n) + \frac{\tau^2}{2h^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n) + \varepsilon \frac{\tau}{h^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n) + \tau f(x, t)$$

Для подобной конечно-разностной схемы был реализован программный код с применением директив как DVM, так и DVMH с подключением графических сопроцессоров.

Проведены исследования с увеличением числа узлов разбиения по пространству. Исследовалось влияние на эффективность и ускорение изменения параметров N, PPN,

THR, CUDA. В таблице 2 приведены результаты численных экспериментов при следующих значениях параметров: узлов по пространству — 1501, шагов по времени — 7000, шагов по вязкости — 10.

Табл. 2. Уравнение Бюргерса-неявная схема

N	PPN	THR	CUDA	T	S	E
1	8	0	0	8,813	1	1
2	8	0	0	8,189	1,076	0,538
4	8	0	0	7,588	1,161	0,290
8	8	0	0	7,182	1,227	0,153
16	8	0	0	7,35	1,199	0,075
1	1	0	1	12,372	0,712	
1	1	2	0	7,467	1,180	0,590
1	2	4	0	6,459	1,364	0,295
2	2	4	0	6,063	1,454	0,148

Аналогичные результаты приведены в таблице 3 при следующих значениях параметров: узлов по пространству — 15001, шагов по времени — 7000, шагов по вязкости — 10.

Табл. 3. Уравнение Бюргерса-невная схема (15001 узел)

N	PPN	THR	CUDA	T	S	E
1	8	0	0	40,54	1	1
2	8	0	0	24,33	1,666	0,833
4	8	0	0	15,92	2,547	0,637
8	8	0	0	11,26	3,602	0,450
16	8	0	0	9,35	4,337	0,271
32	8	0	0	8,32	4,870	0,152
1	1	2	0	23,55	1,721	0,861
1	2	4	0	14,45	2,805	0,701
2	2	4	0	10,06	4,031	0,504
4	2	4	0	9,48	4,277	0,267
8	2	4	0	8,41	4,820	0,151
1	1	0	1	12,80	3,167	
2	1	0	1	18,25	2,221	

Продолжая увеличивать число узлов сеточного разбиения по пространственной переменной, получаем результаты, представленные в таблице 4 для параметров: узлов по пространству — 150001, шагов по времени — 7000, шагов по вязкости — 10.

Табл. 4. Уравнение Бюргерса-невная схема (150001 узел)

N	PPN	THR	CUDA	T	S	E
1	8	0	0	352,89	1	1
2	8	0	0	184,33	1,914	0,957
4	8	0	0	96,38	3,661	0,915
8	8	0	0	51,23	6,888	0,861
16	8	0	0	29,35	12,023	0,751
32	8	0	0	18,35	19,227	0,600
64	8	0	0	12,80	27,572	0,431
128	8	0	0	10,12	34,874	0,272
1	2	4	0	94,76	3,724	0,931
2	2	4	0	50,52	6,985	0,873
4	2	4	0	29,53	11,950	0,747
8	2	4	0	18,42	19,160	0,599
16	2	4	0	12,96	27,231	0,425
32	2	4	0	10,50	33,599	0,262
1	1	0	1	19,53	18,069	
2	1	0	1	21,23	16,625	

По результатам можно сделать вывод о том, что на грубой сетке использование графических сопроцессоров не только не дает никакого эффекта, но даже замедляет выполнение тестового расчета. При резком увеличении количества узлов расчетной сетки обеспечивается эффективная работа параллельных узлов, при этом подключение

графических сопроцессоров также позволяет резко повысить скорость расчетов.

На рисунках 1 и 2 представлены сводные результаты расчетных тестов для решения линейного уравнения Бюргерса. На рисунке 1 представлены графики зависимости эффективности от увеличения числа узлов вычислительного кластера для различного количества узлов расчетной сетки. На рисунке 2 представлены графики зависимости ускорения от увеличения числа узлов вычислительного кластера для различного количества узлов расчетной сетки. Расчеты ускорения представлены для трех вариантов, четвертый вариант (верхний) соответствует «идеальному» ускорению, когда ускорение происходит во столько раз, сколько процессоров выделено.

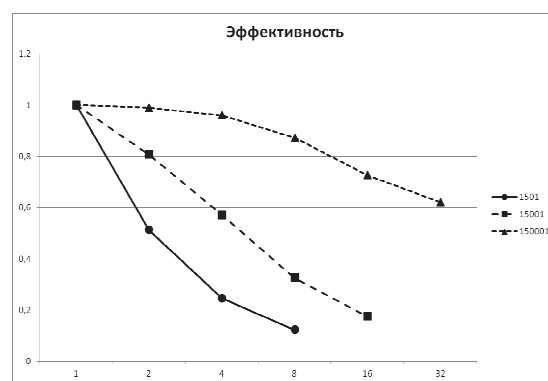


Рис. 1. Значение эффективности при увеличении числа узлов для различного количества узлов расчетной сетки

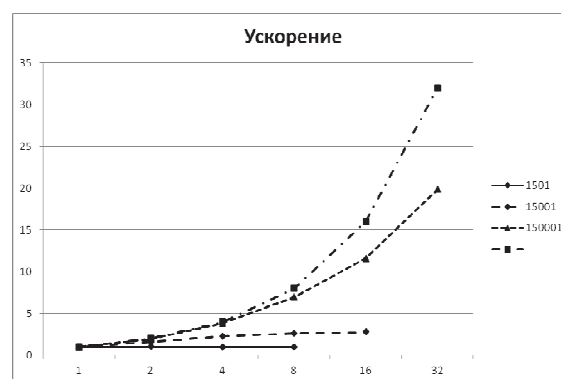


Рис. 2. Значение ускорения при увеличении числа узлов для различного количества узлов расчетной сетки

3 Квазилинейное уравнение Бюргерса

Также тестовые расчеты были проведены для квазилинейного уравнения Бюргерса:

$$\frac{\partial u}{\partial t} + \frac{\partial \left(u \frac{u}{2} \right)}{\partial x} = \varepsilon \frac{\partial^2 u}{\partial x^2}$$

Решением здесь является сглаженная ударная волна:

$$u = \frac{a + b \exp\left(\frac{a-b}{2\varepsilon}(x-x_0-Dt)\right)}{1 + \exp\left(\frac{a-b}{2\varepsilon}(x-x_0-Dt)\right)},$$

где $D = \frac{a+b}{2}$.

Задача решалась при следующем выборе параметров: $a = 1.0$, $b = 0.2$, $\varepsilon = 0.05$, $x_0 = 5.0$.

В качестве сеточного разбиения задавалось: узлов по пространству — 801, шагов по времени— 9000. Аналогично предыдущей задаче проводилось увеличение сеточного разбиения по пространству до 1000001 узлов. Результаты представлены на таблицах 5 и 6.

Табл. 5. Квазилинейное уравнение Бюргерса (801 узел)

N	THR	CUDA	T	S	E
serial			0,5354	1	1
1	0	0	0,6344	0,844	0,844
2	0	0	1,0371	0,5163	0,2581
4	0	0	2,3362	0,2292	0,0573
1	3	0	1,2987	0,4123	0,1374
1	6	0	1,2801	0,4183	0,0697
1	0	1	6,9457	0,0771	
1	0	2	7,2463	0,0739	

Табл. 6. Квазилинейное уравнение Бюргерса (1000001 узел).

N	PPN	THR	CUDA	T	S	E
1	1	0	0	36,75	1	1
2	1	0	0	16,74	2,196	1,098
2	2	0	0	17,25	2,131	1,066
4	2	0	0	9,28	3,962	0,990
4	4	0	0	9,81	3,748	0,937
8	4	0	0	5,86	6,274	0,784
8	8	0	0	8,29	4,433	0,554
1	1	2	0	18,20	2,019	1,010
1	1	4	0	10,60	3,467	0,867
1	1	6	0	8,83	4,162	0,694
1	1	8	0	8,11	4,532	0,567
1	1	0	1	9,83	3,740	
1	1	0	2	9,68	3,798	

Из полученных результатов можно сделать общий вывод. Использование подключения графических сопроцессоров дает резкий эффект ускорения. Но это происходит только в случае достаточно большого количества расчетных узлов. Для грубой сетки, то есть для малого количества расчетных узлов сетки, распараллеливание на несколько процессоров, а тем более подключение графических сопроцессоров приводит к замедлению расчетов по сравнению с последовательным вариантом. Таким образом, расчетная сетка должна быть настолько подробной, чтобы обеспечивать загрузку графических сопроцессоров.

4 Одномерные уравнения газовой динамики

Решалась начально-краевая задача для уравнений динамики невязкого газа (уравнения Эйлера). Исходные уравнения для одномерного течения записывались в консервативной форме:

$$\begin{aligned} \frac{\partial u}{\partial t} + \nabla \cdot f(u) &= 0, \\ x \in \Omega \subset \mathbb{R}, t > 0 \\ x : \Omega \times [0, t] &\rightarrow \mathbb{R} \end{aligned}$$

Где

$u = (\rho, \rho u, \rho E)^T$, $f = (\rho u, \rho u^2 + p, (\rho E + p)u)^T$.
Здесь $\rho E = \rho u^2 / 2 + \rho \varepsilon$ – плотность полной энергии газа, ε – внутренняя энергия газа, $p = \rho \varepsilon (\gamma - 1)$ – давление газа, $\gamma = 1.4$ – показатель адиабаты газа.

В подобной постановке данная задача является классической задачей Сода [11,12], для которой есть точное решение, получаемое итерационным методом. Вектор начальных условий:

$$(\rho, u, p)^T = \begin{cases} (1.0, 0.75, 0.1)^T, & x \leq 2.0 \\ (0.125, 0.0, 0.1)^T, & x > 2.0 \end{cases}$$

Рассматривается отрезок [0,8], время расчета $t = 2.4$, шаг по пространству $h = 0.001$, шаг по времени $\tau = 0.0002$.

Для решения данной системы в используемом программном коде применяются схема типа WENO (взвешенные, существенно не осциллирующие схемы), имеющая 5 порядок аппроксимации по простран-

ственной переменной и 3-й порядок аппроксимации по времени.

Результаты численных экспериментов представлены в таблице 7.

Табл. 7. Задача Сода

N	PPN	THR	CUDA	T	S	E
serial				246,14	1	1
1	12	0	0	259,46	0,95	0,95
2	12	0	0	133,65	1,84	0,92
4	12	0	0	69,79	3,53	0,88
8	12	0	0	38,82	6,34	0,79
12	12	0	0	27,58	8,93	0,74
24	12	0	0	17,6	13,99	0,58
1	1	4	0	69,34	3,55	0,89
1	1	8	0	38,18	6,45	0,81
2	2	6	0	27,58	8,93	0,74
1	1	0	1	25,75	9,56	
2	2	0	1	29,84	8,25	

На рисунке 3 представлена зависимость ускорения для данной задачи от вариантов проведенных в данной серии численных экспериментов.



Рис.3. Зависимость ускорения для задачи Сода от вариантов проведенных численных экспериментов

По итогам проведенных экспериментов следует заметить, что система DVM является эффективным инструментом, позволяющим полноценно использовать функционал организации параллельных вычислений с минимальными трудозатратами со стороны пользователя. Стоит заметить, что система вынуждена переводить фортрановские коды на C++, чтобы избежать использования платных версий CUDA-Fortran. Это накладывает ограничения на структуру программы, такие как обязательное использование

риге функций и невозможность использования массивов без явного обозначения индексов. Если отбросить все эти проблемы и учесть, что приведенные выше задачи являются не настолько вычислительно сложными, как реальные 3D задачи, можно сказать, что DVM-система хорошо подходит для решения задач газо- и гидродинамики.

По результатам проведенных расчетов можно сделать следующий вывод. Подключение графических ускорителей с помощью технологии DVMH может обеспечивать желаемое ускорение для круга задач, обладающих определенным набором свойств при их численной реализации. Определение этого набора предполагается следующим шагом в текущих исследованиях.

Следует также заметить, что исследования проводились на наборе одномерных задач – линейном и квазилинейном уравнениях Бюргера и задаче Сода. Для получения более точной и достоверной информации предполагается проведение аналогичных численных экспериментов для 2D и 3D задач.

Благодарности

Данная работа выполнена при поддержке Российского фонда фундаментальных исследований (проекты 16-01-00553а и 17-01-00444а).

Работа выполнена с использованием оборудования Центра коллективного пользования сверхвысокопроизводительными вычислительными ресурсами МГУ имени М.В. Ломоносова [13] и гетерогенного вычислительного комплекса K-100 Института прикладной математики им. М.В. Келдыша [14].

Список литературы

- [1] Bondarev A.E, Galaktionov V.A. Parametric Optimizing Analysis of Unsteady Structures and Visualization of Multidimensional Data // International Journal of Modeling, Simulation and Scientific Computing, 2013, V.04, N supp01, 13 p., DOI 10.1142/S1793962313410043.
- [2] Бондарев А.Е. Локализация пространственно-временных структур в пространстве определяющих параметров с применением параллельных вычислений / Известия ЮФУ. Технические науки. – №6, 2012. С.100-105.
- [3] Разработка и тестирование параллельного алгоритма решения задач оптимизационного анализа / Бондарев А.Е. [и др.] // Информационные технологии в проектировании и производстве, 2014, № 3, с. 3-9.
- [4] Bondarev A.E., Galaktionov V.A. Analysis of Space-Time Structures Appearance for Non-Stationary CFD Problems // Proceedings of 15-th International Conference On Computational Science ICCS 2015 Rejkjavik, Iceland, June 01-03 2015, Procedia Computer Science, Volume 51, 2015, Pages 1801–1810.
- [5] Построение и визуальный анализ параллельных решений для оптимизационных и параметрических исследований в вычислительной газовой динамике / Бондарев А.Е. [и др.] // Труды 26-ой Международной конференции по компьютерной графике и зрению 19-23 сентября 2016, Нижний Новгород, с.211-215.
- [6] Разработка ин-струментального программного средства Burgers2 для оптимизации гибридных разностных схем / Бондарев А.Е. [и др.] // Препринты ИПМ им.М.В.Келдыша. 2012. № 53. 12 с.
- [7] Высокоуровневая система параллельного программирования DVM URL: <http://dvm-system.org>. (Дата обращения – 01.03.18)
- [8] Расширение DVM-модели параллельного программирования для кластеров с гетерогенными узлами / Бахтин В.А. [и др.] // Супервычисления и математическое моделирование. Труды XIII Международного семинара / Под ред. Р.М. Шагалиева. – Саров: ФГУП “РФЯЦ-ВНИИЭФ”, 2012, с. 84-91.
- [9] Использование языка Fortran DVMH для решения задач гидродинамики на высокопроизводительных гибридных вычислительных системах / Бахтин В.А. [и др.] // Вестник Южно-Уральского государственного университета, серия "Вычислительная математика и информатика", том №2, выпуск №3 – Челябинск: Издательский центр ЮУрГУ, 2013, с 106-120.
- [10] Автоматизация распараллеливания программных комплексов / Бахтин В.А. [и др.] // Научный сервис в сети Интернет: труды XVIII Всероссийской научной конференции (19-24 сентября 2016 г., г. Новороссийск), М.: ИПМ им. М.В.Келдыша, 2016, С. 76-85.
- [11] Sod G.A. Survey of Several Finite Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws // J. Comput. Phys. - 1978. - No. 27. - P. 1–31.
- [12] Toro E. F. Riemann Solvers and Numerical Methods for Fluid Dynamics. - Springer-Verlag, 1999. - 686 p.
- [13] Практика суперкомпьютера "Ломоносов" / Воеводин В.А. [и др.] // Открытые системы. - Москва: Издательский дом "Открытые системы", N 7, 2012. С. 36-39.
- [14] Гибридный вычислительный кластер K-100. URL: <http://www.kiam.ru/MVS/resources> (Дата обращения – 01.03.18)