

## **Решение задачи схлопывания каверны на поверхности жидкости в трехмерной постановке на кластере «СКИФ Cyberia»**

*В. В. Мальшенко, Д. Р. Авзалов*

Кемеровский государственный университет, Кемерово

Впервые задача схлопывания каверны на свободной поверхности под действием силы тяжести была поставлена М. А. Лаврентьевым. В двумерной плоской постановке данная задача в идеальной жидкости решалась многими авторами (В. К. Кедринский, А. Г. Терентьев и др.). Далее в работах К. Е. Афанасьева, Г. Г. Короткова рассматривалась эта же задача, но уже в осесимметричной постановке [2]. В настоящей работе рассматривается задача об эволюции выемки на свободной поверхности в трехмерной постановке.

Для решения поставленной задачи используется метод частиц, разработанный А. М. Франком [1]. Метод частиц основан на свободно-лагранжевом подходе, что позволяет легко отслеживать границы раздела, причем произвольное изменение связности течения и границ не доставляет никаких дополнительных алгоритмических сложностей.

В связи с тем, что решение задач гидродинамики со свободными границами в трехмерной постановке требует больших вычислительных высокопроизводительных ресурсов, для численного моделирования применяется параллельный метод частиц.

Данная работа направлена на исследование вычислительных свойств метода частиц, его распараллеливание и оценку качества параллельного алгоритма.

### **Постановка задачи**

Рассматривается течение идеальной несжимаемой невязкой жидкости в области  $V$ , ограниченной свободной поверхностью  $C_3$  с полусферической выемкой радиуса  $R$ , боковыми твердыми стенками  $C_1$  и непротикаемым дном  $C_2$ , расположенным на глубине  $H$ .

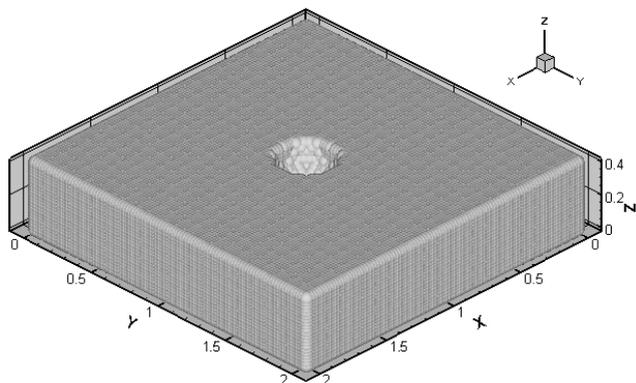


Рис. 1. Начальная геометрия эволюции полусферической выемки над ровным дном

В области  $V$  решается уравнение Эйлера:

$$\frac{r}{dt} = \frac{u}{r}, rOV. \quad (1)$$

Условие несжимаемости жидкости и условие непротекания на твердых стенках:

$$div u = 0, rOV, \quad (2)$$

$$u_n = 0, rOC_2 \quad (3)$$

где  $n$  – внешняя нормаль к границе жидкости.

Начальное поле скоростей  $u_0 = (u_0, v_0) \in (0, 0)$ .

Распараллеливание алгоритма метода частиц.

Основной задачей распараллеливания алгоритма метода частиц является:

- определение количества вычислительных операций;
- определение, какие вычислительные операции могут выполняться независимо друг от друга;
- распределение операций по узлам многопроцессорного ресурса;
- определение необходимых коммуникаций между узлами многопроцессорного ресурса и вычисление объемов передаваемых данных.

Метод частиц подробно описан в [1]. Алгоритм метода частиц, распараллеливание которого выполняется, описан в [6].

Определим количество операций, выполняемое на каждом шаге по времени.

### **Задание начальных условий**

Пусть степень В-сплайна равна 2, количество частиц в области  $N=200\ 000$ , и задана сетка из  $M=m_x \times m_y \times m_z = 30 \times 30 \times 30$  базисных функций.

### **Предиктор**

В начале шага по времени, вычисляется предиктор. На вычисление предиктора для одной частицы в трехмерном пространстве затрачиваем 9 арифметических операций, тогда на весь блок необходимо затратить  $9N$  операций.

### **Вычисление дополнительного блока**

Квадратная матрица  $Mv(i, k)$  размера  $N \times 18$  содержит значения  $v_i \left( r_k^{n+1/2} \right)$ . Вычисление одного значения базисной функции требуют  $t=232$  операций, из них для вычисления В-сплайна второго порядка необходимо 59 операций, для производной необходимо 57 операций. Тогда, для формирования матрицы  $Mv(i, k)$  необходимо  $N \times 18 \times 232 = N \times 4176$  операций.

### **Вычисление матрицы Грамма**

Для вычисления вклада одной частицы в значение одного элемента матрицы необходимо вычислить значение базисной функции в частице и выполнить скалярное произведение, что в трехмерном пространстве даст  $2x+7$  арифметические операции. Далее требуется просуммировать найденные значения по  $k$ . Таким образом, для вычисления одного  $a_{ij}$  необходимо  $(2x+7)N$  операций. Всего перебирается  $M^2$  вариантов. Поэтому для вычисления матрицы  $\{a_{ij}\}$  затрачивается  $(2x+7)NxM^2$  операции.

### Определение правой части СЛАУ

Для вычисления одного элемента правой части в первом алгоритме требуется  $(t + 7 + 9)N$  операций. Тогда для всей правой части  $(t + 7 + 9)NM$ .

### Решение СЛАУ и определение новых скоростей частиц

Решение СЛАУ методом Гаусса с выбором ведущего элемента требует для обоих алгоритмов  $M^3$  операций.

### Вычисление скорости на промежуточном шаге по времени

При определении скорости  $S$  вычисляются значения  $M$  базисных функций, для этого нужно затратить  $t \times M$  операций, затем, умножая вектора на скаляры в трехмерном случае, получаем еще  $5M$  операций. Тогда для вычисления скоростей  $S$  для всех частиц требуется  $N \times M(t + 5)$  арифметических операций.

### Определение новых скоростей и координат частиц

Вычисляются скорости частиц на  $n + 1$  шаге. Число операций требуемых равно  $9N$ . На определение новых координат частиц затрачивается еще  $9N$  операций.

### Общая вычислительная сложность алгоритма

В результате определено количество выполняемых операций за один шаг по времени:

$$T = 9N + (2x + 7)NxM^2 + (t + 7 + 9)NM + M^3 + \\ + NxM(t + 5) + 2x9N = 1,86941E + 13$$

### Декомпозиция вычислительных операций

Проведем декомпозицию каждого блока алгоритма метода частиц. Операции во всех блоках, кроме «решения СЛАУ», имеют идеальный параллелизм, поэтому вычисления в каждом блоке могут быть выполнены независимо друг от друга на SIZE процессорах.

Основной задачей декомпозиции является распределение нагрузки по процессорам таким образом, чтобы минимизировать количество пересылок и увеличить число параллельно выполняемых операций.

Блок «Координаты и скорости на целом шаге» хранит на каждом процессора часть массива координат  $ChCoor(N_p)$  и часть массива скоростей  $ChSkor(N_p)$ . Пересылок не требуется.

Блок «вычисление предиктора» вычисляет на дробном шаге  $n + 1/2$  значение массива координат  $ChCN12(N_p)$ . Пересылок не требуется.

Блок «формирование матрицы СЛАУ». В данном блоке формируется матрица  $A$ , учитывая, предложенные выше механизмы удаления ненужных строк не являются самыми эффективными. Например, при решении данной задачи размер матрицы  $A$  изменяется от 3700 до 4200 в зависимости от характера течения. На каждом процессоре сформирована матрица  $A$ , состоящая из частичных сумм, собираются все части матрицы на одном процессоре, а потом раздаются остальным процессорам. Количество пересылок равно  $M$ , следовательно, время, затраченное на пересылки  $t_s M + t_w M^2$ .

Блок «формирование правой части» может выполняться одновременно с предыдущим блоком, так как полностью от него независим. При учете того, что количество процессоров во время расчета остается числом неизменным, то более эффективной будет реализация параллельного метода частиц, при которой данный блок выполняется после блока «формирование матрицы СЛАУ». Аналогично предыдущему блоку, вектор правой части  $f$  суммируется на одном процессоре и раздается остальным. Количество пересылок равно 1, следовательно, время, затраченное на пересылки  $t_s + t_w M$ .

Блок «решение СЛАУ». Для решения СЛАУ применяется параллельный метод Гаусса с выбором ведущего элемента. Количество пересылок равно  $M$ , время, затраченное на пересылки  $t_s M + t_w \frac{M^2}{2}$ .

Блок «нахождение скорости на дробном шаге». Каждый процессор после выполнения предыдущего блока хранит весь вектор коэффициентов разложения скорости на дробном шаге по базису

пространства, поэтому каждый процессор вычисляет свою часть массива скоростей  $ChUckr(N_p)$ . Пересылка не требуется.

Блок «нахождение новых координат и скорости» каждой процессор вычисляет часть массива координат  $ChCoor(N_p)$  и часть массива скоростей  $ChSkor(N_p)$ . Пересылка не требуется.

Далее осуществляется переход на следующую итерацию.

### Определение теоретического ускорения распараллеленной программы

Для подсчета теоретического ускорения  $S_p = T_1 / T_p$  нужно определить время выполнения одного шага по времени (задачи) на одном процессоре и на  $p$  процессорах.

Проведем теоретическую оценку ускорения и эффективности параллельного алгоритма метода частиц. Сделаем предположение о соотношении времени выполнения одной арифметической операции ко времени выполнения пересылки одной переменной длиной 8 байт:  $c = t_a / t_w$ . Если для объединения узлов в кластере используется сети типа Infiniband или Quadrics, характеризующиеся высокой скоростью передачи данных 800 МБ/с и низкой латентностью менее 8 мкс, то величина  $c = 100$ . Если в качестве сетевой инфраструктуры используется сеть типа GigabitEthernet, характеризующаяся с скоростью передачи данных 300 МБ/с и латентностью более 100 мкс, то величина  $c = 1000$ .

В табл. 1 и 2 представлены теоретические ускорения и эффективность параллельного алгоритма для разных типов сетевой инфраструктуры кластеров при условии одинаковой производительности процессоров.

Таблица 1

*Ускорение и эффективность (теоретические)*

Количество процессоров	Ускорение $c = 100$	Ускорение $c = 1000$	Эффективность $c = 100$	Эффективность $c = 1000$
2	1,88871	1,28855	94,44 %	64,42 %
4	3,57521	1,90008	89,38 %	47,50 %
8	6,45894	2,49124	80,74 %	31,14 %
16	10,82435	2,95016	67,65 %	18,43 %
32	16,3493	3,24947	51,09 %	10,15 %
64	21,9517	3,42311	34,30 %	5,34 %

Полученные теоретические оценки позволяют охарактеризовать параллельный алгоритм метода. Алгоритм является сильно связанным, поэтому изменение характеристик сети сильно сказывается на эффективности параллельного алгоритма. В результате чего можно сделать вывод о том, что данный подход к распараллеливанию сильно ограничивает круг многопроцессорных вычислительных кластеров, на которых можно производить расчеты задач решаемых данным методом.

### **Результаты распараллеливания алгоритма метода частиц**

Для проведения численных расчетов использовался кластер «СКИФ Siberia» Томского государственного университета. В расчетах использовалось от 1 до 64 процессоров.

Таблица 2  
*Ускорение и эффективность (экспериментальные)*

Количество процессоров	Время, сек	Ускорение	Эффективность
1	30,5625	—	—
2	12,8092	2,3860	119,29%
4	7,4191	4,1194	102,98%
8	3,1271	9,7734	122,16%
16	1,4908	20,5010	128,13%
32	1,0712	28,5305	89,15%
64	0,9273	32,9574	51,49%

В табл. 2 показана зависимость времени выполнения одного итерационного шага задачи от количества процессоров, на которых выполнялся расчет. Анализируя полученные ускорение и эффективность параллельной задачи можно сделать вывод о хорошем качестве распараллеливания алгоритма метода частиц для решения задач гидродинамики со свободными границами.

Анализируя ускорение, можно сделать вывод о том, что для решения поставленной задачи с заданным набором параметров не целесообразно использование более 16 процессоров, так как время, затрачиваемое на пересылки данных, становится преобладающим над временем, затрачиваемым на выполнение вычислительных операций.

Высокая эффективность параллельного алгоритма является следствием того, что с увеличением количества процессоров, задействованных в расчете, уменьшается объем оперативной памяти, используемой на каждом процессоре, это приводит к эффективному использованию кэш-память процессора, уменьшению кэш-промахов, и как результат, уменьшение времени выполнения арифметических операций.

Результаты распараллеливания программы позволяют сделать следующие выводы:

Параллельный вариант программы решения задач гидродинамики со свободными границами позволяет решать задачи большей размерности за счет использования больших объемов оперативной памяти; решение задачи на кластере «СКИФ Cyberia» позволяет существенно сократить время численных расчетов; для поставленной задачи эффективным является использование 16 процессоров.

### Литература

1. **Франк А. М.** Дискретные модели несжимаемой жидкости. — М. : Физматлит, 2001.
2. **Afanas'ev K. E., Korotkov G. G.** Эволюция полукруговой выемки на свободной поверхности в плоском и осесимметричном случаях // The International Summer Scientific School «High Speed Hydrodynamics», June 2002, Cheboksary.
3. **Афанасьев К. Е., Стуколов С. В.** КМГЭ для решения плоских задач гидродинамики и его реализация на параллельных компьютерах : Учеб. пособие. — Кемерово : КемГУ, 2001. — 208 с.
4. **Овсянников Л. В.** Общие уравнения и примеры // Задачи о неустановившемся движении жидкости со свободной границей. — Новосибирск : Наука, 1967. — С. 5—75.
5. **Завьялов Ю. С., Квасов Б. И., Мирошниченко В. Л.** Методы сплайн-функций. — М. : Наука, 1980. — 352 с.
6. **Мальшенко В. В.** Параллельная реализация метода частиц для решения задач гидродинамики со свободными границами // Третья Сибирская школа-семинар по параллельным вычислениям / Под ред. проф. А. В. Старченко. — Томск : Изд-во Том. ун-та, 2006. — С. 50—60.