

# БАЛАНСИРОВКА ВЫЧИСЛЕНИЙ НА ПАРАЛЛЕЛЬНЫХ ЭВМ В МЕТОДЕ ЧАСТИЦ В ЯЧЕЙКЕ РАСЧЕТА ЗАДАЧ ГАЗОВОЙ ДИНАМИКИ

*Н. В. Мелешкин, С. Н. Полищук, М. А. Сизова, А. А. Уткина*

ФГУП «РФЯЦ-ВНИИЭФ», г. Саров Нижегородской обл.

В настоящее время для решения нестационарных задач механики сплошной среды создан ряд численных методов, каждый из которых эффективен для определенного круга задач. Для моделирования течения с большими деформациями, приводящими к фрагментации вещества, расчетные методики строятся, как правило, на основе метода дискретных частиц [1–9] либо метода концентраций [10, 11]. При использовании метода частиц, в котором каждая частица является носителем информации о вполне конкретном веществе, достаточно просто решается вопрос расчета контактных границ и разрушения материала, включая нарушение сплошности вещества. В данной работе представлено описание методики и численных алгоритмов расчета задач газовой динамики с теплопроводностью на прямоугольной адаптивно-встраиваемой сетке методом частиц в ячейке на параллельных ЭВМ. Распараллеливание методики основано на геометрической декомпозиции расчетной области и использовании неблокирующих обменов. Геометрия расчетной области определяется параллелепипедом, в котором расположены частицы с заданными координатами и физическими величинами. Частицы, в большинстве случаев, расположены неравномерно по пространству. Для сбалансированной загрузки процессоров используется декомпозиция задачи на параллелепипеды с учетом неравномерности расположения частиц в расчетной области.

## 1. Описание методики

Метод частиц в ячейке («Particle In Cell») был предложен Харлоу в 1955 году [1]. Впоследствии этот метод получил дальнейшее развитие в ряде работ [2–9] и применялся для решения широкого класса задач. В методе Харлоу среда описывается системой точечных частиц конечной массы. Расчет ведется на неподвижной эйлеровой сетке, покрывающей область движения среды. Параметры поля течения в эйлеровых ячейках определяются осреднением по частицам, находящимся в ячейке. Дифференциальные уравнения расщепляются на две вспомогательные системы (1), (2), отвечающие двум этапам в расчетном алгоритме. На первом, эйлеровом, этапе учитывается изменение параметров за счет сил давления. На втором, лагранжевом, этапе с помощью модели частиц учитываются процессы переноса.

$$\left\{ \begin{array}{l} \frac{\partial p}{\partial t} = 0, \\ \frac{\partial p_\alpha}{\partial t} = 0, \\ \frac{\partial \rho u}{\partial t} + \frac{\partial p}{\partial x} = 0, \\ \frac{\partial \rho v}{\partial t} + \frac{\partial p}{\partial y} = 0, \\ \frac{\partial \rho w}{\partial t} + \frac{\partial p}{\partial z} = 0, \\ \frac{\partial \rho E}{\partial t} + \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} = 0. \end{array} \right. \quad (1)$$

$$\left\{ \begin{array}{l} \frac{\partial p}{\partial t} = 0, \\ \frac{\partial p_\alpha}{\partial t} = 0, \\ \frac{\partial \rho u}{\partial t} = 0, \\ \frac{\partial \rho v}{\partial t} = 0, \\ \frac{\partial \rho w}{\partial t} = 0, \\ \frac{\partial \rho E}{\partial t} = \frac{\partial}{\partial x} \left( \chi \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( \chi \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( \chi \frac{\partial T}{\partial z} \right) + \rho Q(x, y, z, t). \end{array} \right. \quad (2)$$

Расчет одного шага метода частиц в ячейке состоит из следующих вычислений. На первом этапе строится адаптивная эйлерова сетка. Ячейки базовой сетки, в которых число частиц больше заданного числа, разбиваются на  $2^v$  подобных ячеек, где  $v$  – размерность задачи. Дробление ячеек ограничивается вложенностью сетки и числом частиц в ячейке. Затем устанавливаются связи между частицами и ячейками, а также связи между соседними ячейками. Определяются средние значения расчетных параметров в ячейках адаптивной сетки, и определяется величина шага интегрирования по времени уравнений газовой динамики. После этого решается система уравнений (1) и определяются новые значения компонент вектора скорости и внутренней энергии за счет сил давления. В конце первого этапа вычисляется изменение внутренней энергии и температуры за счет теплопроводности. Решение уравнения теплопроводности (2) реализовано по двум разностным

схемат: явной и неявной. Решение уравнения теплопроводности по явной разностной схеме выполняется со своим шагом по времени, который зачастую на несколько порядков меньше газодинамического. При решении уравнения теплопроводности по неявной разностной схеме расчетный шаг полагается равным газодинамическому шагу. На втором (завершающем) этапе определяются значения компонент вектора скорости, внутренней энергии и температуры для каждой частицы в ячейке и новое положение частиц. Для контроля расчета на заключительном этапе определяются интегральные характеристики системы (интегралы массы, импульсов, полной и внутренней и кинетической энергии), а также потоки массы импульсов, энергии и работы через границы области. Этим завершается расчет одного шага.

## 2. Параллельная реализация

Распараллеливание методики основано на геометрической декомпозиции расчетной области и использовании неблокирующих обменов. Геометрия расчетной области определяется параллелепипедом, в котором расположены частицы с заданными координатами и физическими величинами. Частицы, в большинстве случаев, расположены неравномерно по пространству. Для сбалансированной загрузки процессоров используется декомпозиция задачи на параллелепипеды с учетом неравномерности расположения частиц в расчетной области. При расчете в многопроцессорном режиме используются два типа обменов: обмены сеточными величинами, отнесенными к адаптивным ячейкам, и обмены частицами. Распараллеливание осуществляется средствами параллельной библиотеки MPI.

### 2.1. Установление межпроцессорного соседства

Расчетная область представляет собой параллелепипед, который равномерно разбивается на линии вдоль осей  $Ox$ ,  $Oy$  и  $Oz$ . Сетку, полученную пересечением данных линий, будем называть базовой. После построения базовой сетки осуществляется построение адаптивно-встраиваемой сетки. Критерием дробления сетки является максимальное число частиц в ячейке; если в ячейке число частиц больше максимально заданного, то ячейка дробится на восемь равных ячеек, после чего для каждой полученной ячейки производится дробление и проверка введенного критерия. В параллельном режиме производится декомпозиция базовой расчетной сетки (рис. 1). Декомпозиция осуществляется с учетом неравномерности расположения частиц (более подробное описание приведено ниже) и сохранения расчетной формы параллелепипеда на процессоре. Ограничение на форму связано с сохранением однородности реализуемых алгоритмов и упрощением алгоритмов пересылки частиц между процессорами.

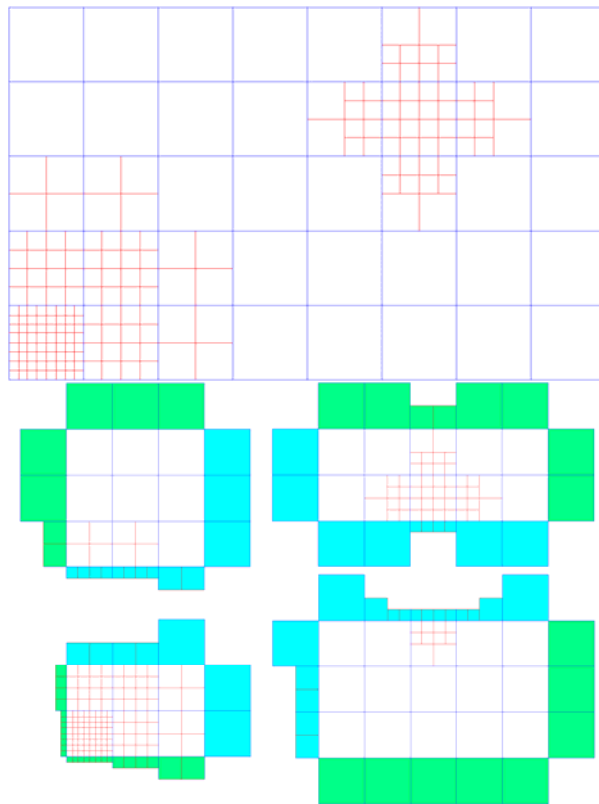


Рис. 1. Декомпозиция расчетной сетки

Расчетная базовая сетка, по построению, является равномерной регулярной сеткой, каждая ячейка которой определяется трехмерным индексом  $(i, j, k)$ . Поэтому на каждом процессоре расчетный блок определяется двумя трехмерными индексами: индексы крайней ближней ячейки  $(i_{\min}, j_{\min}, k_{\min})$  и крайней дальней ячейки  $(i_{\max}, j_{\max}, k_{\max})$  в глобальной нумерации. Установление межпроцессорного соседства и подготовка данных для обменов выполняются по следующему алгоритму. Каждый процессор на границах блока окаймляется слоем обменных базовых ячеек, после чего для каждого процессора формируется список номеров соседних процессоров (процессор является соседним, если его базовые ячейки являются обменными на текущем процессоре). Процессор для каждого соседнего формирует списки «send» базовых ячеек (посылаемые ячейки, которые являются счетными на текущем процессоре и обменными на соседнем) и «recv» базовых ячеек (принимаемые ячейки, которые являются обменными на текущем процессоре и счетными на соседнем). С учетом подготовленных «send» и «recv» списков базовых ячеек формируются списки «send» и «recv» адаптивных ячеек, грани которых являются смежными с гранями базовых счетных ячеек. В завершение на гранях процессора, смежных с расчетной областью, добавляется слой фиктивных (граничных) ячеек для реализации граничных условий. При формировании адаптивных ячеек по описанному алгоритму вводится следующая нумерация: сначала нумеруются счетные ячейки, потом по каждому соседнему процессору –

обменные ячейки, а затем – граничные. Данная сеточная нумерация позволяет вводить сквозную нумерацию ячеек на процессорах без введения дополнительной перенумерации.

## 2.2. Реализация межпроцессорных обменов

В процессе расчета используются два типа обменов: обмены сеточными величинами, отнесенными к адаптивным ячейкам, и обмены частицами. По сформированным спискам номеров соседних процессоров и информации о межпроцессорных связях номеров «*send*» и «*recv*» адаптивных ячеек процессоров осуществляется первый тип парных обменов. Последовательность пересылок между процессорами выполняется по правилу «старший -> младшему»: процессор с большим идентификатором посылает информацию процессорам с меньшим идентификатором и ждет от него пересылки, и наоборот: процессор с меньшим идентификатором сначала ждет, а затем посылает. Данное правило гарантирует неблокирующие друг друга обмены.

Второй тип обменов (обмен частицами) состоит из следующих шагов:

- выполняется проверка на выход за границы процессорного блока частиц по направлению  $Ox$ ;

в случае если частица вылетела за границы блока, производится ее добавление в буфер отправки соседнему процессору; поскольку у процессора с одной стороны может быть несколько соседей, то при определении номера принимающего процессора выполняется проверка координат частицы по осям  $Oy$  и  $Oz$ ;

- выполняется пересылка частиц по направлению  $Ox$  (обмены осуществляются по правилу «старший -> младшему»);

- выполняются проверка на выход за границы процессорного блока частиц по направлению  $Oy$  и формирование буфера отправки, при этом в проверку включаются и частицы, появившиеся на процессоре при пересылке по направлению  $Ox$ ;

- выполняется пересылка частиц по направлению  $Oy$ ;

- выполняется проверка на выход за границы процессорного блока частиц по направлению  $Oz$  и формирование буфера отправки, при этом в проверку включаются и частицы, появившиеся на процессоре при пересылке по направлениям  $Ox$  и  $Oy$ ;

- выполняется пересылка частиц по направлению  $Oz$ .

Использование данного алгоритма пересылки частиц позволяет пересылать информацию о частицах не только через грань процессорного блока, но и через ребро, и через узел блока (рис. 2).

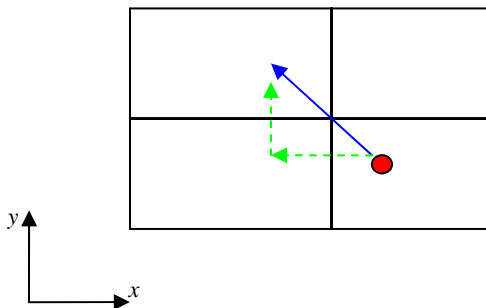


Рис. 2. Схема пересылки частицы через узел процессорного блока

После завершения пересылок производится перепорядочивание массивов частиц в связи с тем, что некоторые частицы могли вылететь за границы процессорного блока, а некоторые – прилететь.

## 3. Описание алгоритмов балансировки

Декомпозиция расчетной области проводится по координатным линиям базовой сетки. После расчета начальных данных по заданному разбиению на процессоры (количество процессоров по осям  $Ox$ ,  $Oy$  и  $Oz$ ) производится последовательная декомпозиция расчетной области, сначала на слои базовых ячеек по первому направлению, по полученным слоям производится нарезка слоев ячеек по второму направлению и затем по третьему (рис. 3). В данной работе предлагается проводить балансировку по значениям максимального уровня дробления базовых ячеек. Выбор данного критерия обосновывается подходом к построению адаптивной сетки. Если в ячейке число частиц больше максимально заданного значения, то производится дробление ячейки на 8 равных. Уровень дробления выступает в качестве параметра нагрузки вычисления на базовую ячейку в предположении равномерного заполнения ячейки частицами.

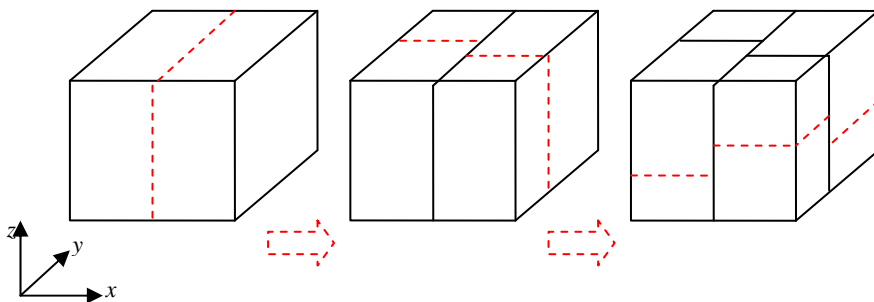


Рис. 3. Последовательная декомпозиция расчетной области

Опишем более подробно алгоритм балансировки по одному из направлений. В качестве алгоритма балансировки по заданному направлению предлагается использовать «эвристический алгоритм», описанный в работе [9]. Для обобщения алгоритма введем понятие меры нагрузки базовой ячейки, в текущей работе в качестве меры будет выступать максимальный уровень дробления ячейки, полученный при построении.

Введем следующие обозначения:  $nCount$  – количество слоев базовых ячеек,  $pCount$  – массив значений меры в слое базовых ячеек,  $pDiv$  – распределе-

ние слоев по процессорам,  $nAvgCount$  – среднее количество частиц на процессоре. На рис. 4 представлена блок-схема алгоритма.

При работе данного алгоритма сначала определяется граница 0-го процессора, затем в цикле по элементам массива  $pCount$ , слой  $i + 1$  назначается на процессор с номером 1, если при этом величина меры в первых  $id + 1$  процессорах менее отличается от  $(id + 1) * nAvgCount$ , чем без величины меры этого слоя. Данный «эвристический алгоритм» позволяет строить достаточно хорошую загрузку процессоров по выбранной мере.

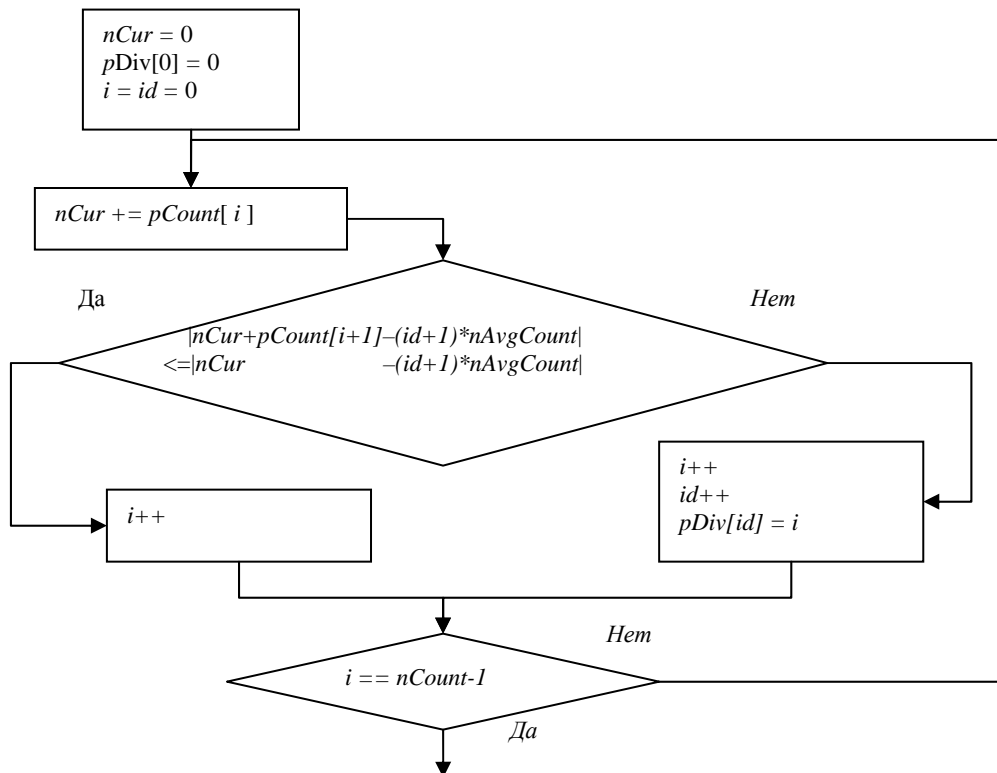


Рис. 4. Блок-схема алгоритма балансировки по слоям

Данный алгоритм применяется как непосредственно перед началом расчета, так и в процессе его проведения. В качестве критерия балансировки предлагается использовать максимальное допустимое значение дисбаланса меры (максимальный уровень дробления базовой ячейки). На каждом  $i$ -м процессоре производится вычисление суммарного значения максимально построенного уровня дробления по всем  $j$ -м базовым ячейкам

$$nSumLayer_i = \sum_j nLayer_j. \quad (3)$$

Тогда дисбаланс максимального уровня дробления по процессорам вычисляется по следующей формуле:

$$dis = \frac{\max_i(nSumLayer_i) - \langle nSumLayer_i \rangle}{\max_i(nSumLayer_i)} \times 100\%, \quad (4)$$

где  $\max_i(nSumLayer_i)$  – максимальное суммарное количество уровней дробления по процессорам,  $\langle nSumLayer_i \rangle$  – среднее суммарное количество уровней дробления по процессорам, вычисляемое по формуле

$$\langle nSumLayer_i \rangle = \sum_i \frac{nSumLayer_i}{N}, \quad (5)$$

где  $N$  – число процессоров.

При достижении максимально заданного значения дисбаланса уровня производится новая декомпозиция задачи и перераспределение частиц по процессорам.

## Результаты тестирования

Задача о точечном взрыве. Рассматривается одномерная сферическая задача о точечном взрыве

в однородном идеальном газе [12]. В сфере радиусом  $r = 0,1$  задана начальная внутренняя энергия  $\varepsilon = 10^7$ , а в сферическом слое  $0,1 < r < 20$  начальная внутренняя энергия равна 0. Плотность  $\rho$  в начальный момент времени в обеих областях равна 1. Модель среды – идеальный газ с показателем адиабаты 1,4, газ находится в состоянии покоя. Расчеты проводились восьмой части задачи  $0 \leq x, y, z \leq 20$  до момента времени  $t = 4$ . В области энерговыделения плотность расстановки частиц была максимальной и далее ступенчато уменьшалась. Общее количество частиц было задано равным 1006168, количество базовых ячеек по каждому координатному направлению задано равным 50. В табл. 1 представлена зависимость времени счета задачи от числа процессоров в двух режимах решения задачи: без балансировки (равномерная декомпозиция) и с динамической балансировкой.

Пробитие шаром преграды. Рассматривается задача пробития шаром преграды [13]. Расчетная область представлена на рис. 5. Толщина пластины равна 0,9 см, радиус шара – 0,5 см. В расчете использовалось уравнение состояния идеального газа:

$P = (\gamma - 1)E\rho$ ,  $\gamma = 3$ . В области 1 задано  $P = 0$  ГПа,  $\rho = 10$  г/см<sup>3</sup>,  $U = -1$  км/с,  $V = 0$  км/с, в области 2  $P = 0$  ГПа,  $\rho = 1$  г/см<sup>3</sup>,  $U = 0$  км/с,  $V = 0$  км/с.

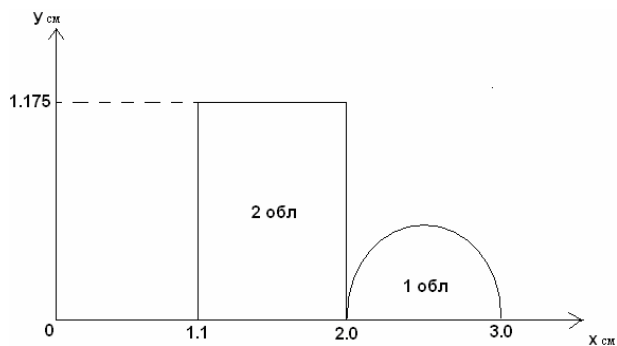


Рис. 5. Расчетная область

Общее число частиц в расчете задано равным 1928126. В табл. 2 представлена зависимость времени счета задачи от числа процессоров в двух режимах решения задачи: без балансировки и с динамической балансировкой (максимальный дисбаланс уровней дробления задан равным 25 %).

Таблица 1

Сравнение общего времени счета задачи

| Число процессоров | Число процессоров по оси $Ox$ | Число процессоров по оси $Oy$ | Число процессоров по оси $Oz$ | Время счета задачи без балансировки $t1$ , мин | Время счета задачи с балансировкой $t2$ , мин | $t1/t2$ |
|-------------------|-------------------------------|-------------------------------|-------------------------------|--|---|---------|
| 1                 | 1                             | 1                             | 1                             | 248  | 248   | 1,00    |
| 2                 | 2                             | 1                             | 1                             | 163  | 139   | 1,17    |
| 3                 | 3                             | 1                             | 1                             | 119  | 101   | 1,18    |
| 4                 | 2                             | 2                             | 1                             | 97   | 75  | 1,29    |
| 6                 | 3                             | 2                             | 1                             | 71   | 53  | 1,34    |
| 8                 | 2                             | 2                             | 2                             | 68   | 44  | 1,55    |
| 8                 | 4                             | 2                             | 1                             | 64   | 41  | 1,56    |

Таблица 2

Сравнение общего времени счета задачи

| Число процессоров | Число процессоров по оси $Ox$ | Число процессоров по оси $Oy$ | Число процессоров по оси $Oz$ | Время счета задачи без балансировки $t1$ , мин | Время счета задачи с балансировкой $t2$ , мин | $t1/t2$ |
|-------------------|-------------------------------|-------------------------------|-------------------------------|--|---|---------|
| 1                 | 1                             | 1                             | 1                             | 320  | 320   | 1,00    |
| 2                 | 2                             | 1                             | 1                             | 201  | 173.3   | 1,16    |
| 3                 | 3                             | 1                             | 1                             | 137,6  | 117.6   | 1,17    |
| 4                 | 2                             | 2                             | 1                             | 115,1  | 90.6  | 1,27    |
| 6                 | 3                             | 2                             | 1                             | 83   | 62.4  | 1,33    |
| 8                 | 2                             | 2                             | 2                             | 78,4   | 50.6  | 1,55    |
| 8                 | 4                             | 2                             | 1                             | 79,3   | 50.8  | 1,56    |

Из приведенных таблиц видно, что алгоритмы балансировки на основе выравнивания меры максимального уровня дробления базовых ячеек позволяют ускорять время расчета задачи ~ 1,3–1,5 раза.

## Литература

1. Харлоу Ф. Х. Численные методы частиц в ячейках для задач гидродинамики // Вычислительные методы в гидродинамике. М.: Мир, 1967. С. 316–342.

2. Яненко Н. Н., Анучина Н. Н., Петренко В. Е., Шокин Ю. И. О методах расчета задач газовой динамики с большими деформациями // Сб. «Численные методы механики сплошной среды». Новосибирск, 1970. Т. 1, № 1. С. 40–62.

3. Анучина Н. Н. О методах расчета течений сжимаемой жидкости с большими деформациями // Там же. № 4. С. 3–84.

4. Павленко О. Н., Литвиненко И. А. Моделирование трехмерного течения газа методом частиц на адаптивно-встраиваемой сетке. Расчет задачи о точечном взрыве // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 2003. Вып. 4. С. 61–67.

5. Белоцерковский О. М., Давыдов Ю. М. Не-стационарный метод «крупных частиц» для газодинамических расчетов // Вычислительная математика и математическая физика. 1971. Т. 11, № 1. С. 182–207.

6. Агурейкин В. А., Крюков Б. П. Метод индивидуальных частиц для расчета течений многокомпонентных сред с большими деформациями // Сб. «Числ. методы механики сплошной среды». Новосибирск, 1986. Т. 17, № 1. С. 17.

7. Башуров В. В., Пятянина А. В. Численный метод сглаженных частиц в трехмерной постановке для моделирования физических процессов на примере задач столкновения и пробивания // Вопросы атомной науки и техники. Сер. Математическое моделирование физических процессов. 1999. Вып. 1. С. 26–38.

8. Адрианов А. Н., Ефимкин К. Н. Подход к параллельной реализации метода частиц в ячейках // Препринты ИПМ им. М. В. Келдыша, 2009.

9. Малышкин В. Э., Корнеев В. Д. Параллельное программирование мультимедийных компьютеров. Новосибирск, 2006. С. 185–237.

10. Бахрах С. М., Глаголева Ю. П., Самигулин С. М. и др. // ДАН СССР. 1981. Т. 257, № 3. С. 556.

11. Бахрах С. М., Спиридонов В. Ф., Шанин А. А. // Там же. 1984. Т. 278. Вып. 4. С. 556.

12. Седов Л. И. Методы подобия и размерности в механике. М.: Наука, 1981. С. 243–274.

13. Результаты расчетов некоторых производственно-методических задач газовой динамики по модифицированному методу ИНЧ: Отчет о НИР. Предприятие п/я Г – 4665, Р. М. Шагалиев; инв. № 8/10354, 1989.