

Модификация параллельного алгоритма для численного моделирования открытой магнитной ловушки

М. А. Боронина¹, В. А. Вшивков¹, С. Е. Киреев^{1,2}

¹ *Институт вычислительной математики и математической геофизики СО РАН
Новосибирск, Россия*

² *Новосибирский государственный университет
Новосибирск, Россия*

Аннотация

Работа посвящена численному моделированию динамики плазмы в диамагнитном режиме в открытой магнитной ловушке. Используемая гибридная модель основана на кинетическом описании ионной компоненты и магнитно-гидродинамическом описании электронной компоненты плазмы. Недостатком модели является условная устойчивость и соответствующие требования к временному шагу. На практике измельчение сетки в два раза требует уменьшения временного шага в 6 раз. Для характерных времен плазменных процессов порядка 102 обратных ионных циклотронных частот вычислений на сетке 100×500 требует несколько дней. При этом более 85 % времени работы программы занимают процедуры обработки частиц, поэтому их эффективная реализация играет ключевую роль в сокращении времени расчетов. В предлагаемом вниманию алгоритме мы сочетаем динамическую загрузку балансировки и векторизацию вычислений расчета плотности электронов и плотности токов. Представлены результаты численных экспериментов с учетом существенно неравномерного распределения частиц в области и увеличивающегося их числа за счет постоянной инжекции.

Ключевые слова

метод частиц-в-ячейках, параллельные алгоритмы, векторизация, численное моделирование, физика плазмы, диамагнитный режим

Благодарности

Концепция гибридной модели создана в рамках госзадания ИВМиМГ СО РАН (проект 0315-2019-0009). Численные эксперименты выполнены при поддержке Российского фонда фундаментальных исследований, проект 18-29-21025.

Для цитирования

Боронина М. А., Вшивков В. А., Киреев С. Е. Модификация параллельного алгоритма для численного моделирования открытой магнитной ловушки // Вестник НГУ. Серия: Информационные технологии. 2021. Т. 19, № 1. С. 15–25. DOI 10.25205/1818-7900-2021-19-1-15-25

Modification of Parallel Algorithm for Numerical Modeling of Open Magnetic Trap

M. A. Boronina, V. A. Vshivkov, S. E. Kireev

¹ *Institute of Computational Mathematics and Mathematical Geophysics SB RAS
Novosibirsk, Russian Federation*

² *Novosibirsk State University
Novosibirsk, Russian Federation*

Abstract

The work is devoted to the numerical simulation of plasma dynamics in the diamagnetic regime in an open magnetic trap. The hybrid model we use is based on the kinetic description of the ion and magnetohydrodynamic description of the electron plasma component. The disadvantage of the model is the conditional stability and the corresponding re-

quirements for the time step. In practice, the double mesh nodes increase requires reducing the time step by factor of 6. For the typical times of the plasma processes of the order of 10^2 reciprocal ion cyclotron frequencies, the calculations on 100×500 grid require several days. At the same time, more than 85 % of the program operation time is occupied by the particle processing procedures, so their efficient implementation plays a key role in the decrease of the calculation times. In the proposed algorithm, we combine dynamic load balancing and vectorization of calculations for the electron density and current density computations. The results of numerical experiments are presented taking into account the substantially nonuniform distribution of the particles in the region and their increasing number due to the constant injection.

Keywords

particle-in-cell method, parallel algorithms, vectorization, numerical modeling, plasma physics, diamagnetic regime

Acknowledgements

The concept of the hybrid model was created within the framework of the state task of the IVMiMG SB RAS (project 0315-2019-0009). The numerical experiments were carried out with the support of the Russian Foundation for Basic Research, project 18-29-21025.

For citation

Boronina M. A., Vshivkov V. A., Kireev S. E. Modification of Parallel Algorithm for Numerical Modeling of Open Magnetic Trap. *Vestnik NSU. Series: Information Technologies*, 2021, vol. 19, no. 1, p. 15–25. (in Russ.) DOI 10.25205/1818-7900-2021-19-1-15-25

Введение

Оригинальная концепция удержания плазмы в линейной магнитной ловушке была предложена в [1]. Идея основана на выталкивании магнитного поля ловушки частицами, инжектируемыми в ловушку. Возле точки инжекции образуется область, имеющая форму пузыря, где значения магнитного поля малы. В тонком граничном слое этого «диамагнитного» пузыря градиент электромагнитного поля высок, большая часть ионов не пересекает границу пузыря. Отношение давления плазмы к давлению магнитного поля достигает значений $\beta \sim 1$. Высокие значения β ($\sim 0,6$ в экспериментах ГДЛ [2]), простота конструирования делает компактные магнитные ловушки перспективной альтернативой токамакам в области управляемого термоядерного синтеза. Вопросы формирования пузыря и его устойчивости рассмотрены теоретически, и предполагается, что создание новой ловушки САТ (ИЯФ СО РАН, Новосибирск, Россия) позволит продемонстрировать удержание плазмы в «диамагнитном» режиме [3].

Численные эксперименты для изучения динамики плазмы в магнитном поле могут быть проведены на основе гибридной модели [4; 5] с использованием метода частиц-в-ячейках [6]. Кинетический подход для ионов и электронов сложен в применении из-за различий в пространственных и временных масштабах из-за отношения масс $m_i/m_e \sim 2 \cdot 10^3$: ларморовский радиус электрона требует пропорционально более мелких сеток и соответствующего увеличения вычислительных затрат.

Гибридное описание, которое включает в себя кинетическое уравнение Власова для ионной компоненты плазмы и магнитно-гидродинамические уравнения для электронной, позволяет снизить требования к ресурсам ЭВМ.

Другим направлением ускорения получения численного решения является распараллеливание кода и проведение вычислений на большом количестве процессоров. Подход, основанный на векторизации, когда одна операция применяется к нескольким данным одновременно, позволяет использовать процессоры в интенсивном режиме и улучшить производительность кода.

В работе мы представляем численный алгоритм, основанный на описанных подходах для проведения вычислений в «диамагнитном» режиме взаимодействия плазмы с электромагнитным полем открытой магнитной ловушки с учетом неравномерного распределения частиц во времени и пространстве.

Математическая модель

Рассматривается 2D-3V модель цилиндрической открытой ловушки, заполненной в начальный момент времени фоновой водородной плазмой плотностью n_0 . Мы используем прямоугольную область размером $[R_{\max} \times L_{\max}]$ и цилиндрическую систему координат. Магнитное поле в центре ловушки $(0, L_{\max}/2)$ равно H_0 , оно определяется двумя катушками с током на концах ловушки, при этом пробочное отношение равно $R_m = 2$. В точке $(0, L_{\max}/2)$ постоянно вводится ионный пучок. Предполагается, что плазма квазинейтральна, плотности ионной и электронной компоненты плазмы совпадают: $n = n_i = n_e$.

В безразмерных переменных траектории ионов описываются характеристиками уравнения Власова:

$$\frac{d\vec{v}_i}{dt} = \vec{F}_i, \quad \frac{d\vec{r}_i}{dt} = \vec{v}_i,$$

где $\vec{v}_i$ – скорости ионов, $\vec{r}_i$ – их координаты, $\vec{F}_i$ – комбинация силы Лоренца и силы трения между ионами и электронами:

$$\vec{F}_i = \vec{E} + \frac{1}{c}[\vec{v}_i, \vec{B}] - k \frac{\vec{j}}{n}, \quad k = cm_e / eH_0 \tau_{ei},$$

где τ_{ei} – характерное время между столкновениями ионов и электронов [7].

Уравнение для вычисления скоростей электронов $\vec{V}_e$:

$$\vec{j} = n_i \vec{V}_i - n_e \vec{V}_e.$$

Средняя скорость ионов $\vec{V}_i$ и ионная плотность n_i являются интегралами функции распределения $f_i(t, \vec{r}, \vec{v})$:

$$n_i(\vec{r}) = \int f_i(t, \vec{r}, \vec{v}) d\vec{v},$$

$$\vec{V}_i(\vec{r}) = \frac{1}{n_i(\vec{r})} \int f_i(t, \vec{r}, \vec{v}) \vec{v} d\vec{v}.$$

Мы используем приближение безмассовых электронов для вычисления электрического поля $\vec{E}$:

$$\vec{E} + [\vec{V}_e, \vec{B}] + \frac{\nabla p_e}{n} - k \frac{\vec{j}}{n} = 0.$$

Температура электронов определяется нагревом вследствие трения между электронами и ионами и изменением за счет расширения / сжатия:

$$n \left(\frac{\partial T_e}{\partial t} + (\vec{V}_e \nabla) T_e \right) = \frac{2}{3} \left(2k \frac{j^2}{n} - p_e \operatorname{div} \vec{V}_e \right),$$

$p_e = n_e T_e$ – давление электронов, $k = \text{const}$.

Предполагается, что токи смещения малы, и для вычисления магнитного поля $\vec{H}$ и токов $\vec{j}$ используются уравнения Максвелла:

$$\frac{\partial \vec{H}}{\partial t} = -\operatorname{rot} \vec{E}, \quad \operatorname{rot} \vec{H} = \vec{j}.$$

В начальный момент электрическое поле $\vec{E} = 0$, магнитное поле определяется уравнением Пуассона для потенциала катушек. Предполагается, что частицы отражаются от стенок цилиндра и вылетают через торцы ловушки. Считается, что возмущения всех величин на границах пренебрежимо малы.

Алгоритмы

Для решения задачи используется метода частиц-в-ячейках: реальные частицы сгруппированы в модельные частицы с координатами, скоростями и зарядами $r, z, v_r, v_\phi, v_z, q$, определёнными на лагранжевой сетке. Остальные величины определены на эйлеровой сетке, при этом используются различные пространственные сетки для различных величин [8]:

$$\begin{aligned} & B_r(i-1/2, k), \quad B_\phi(i, k), \quad B_z(i, k-1/2), \\ & E_r(i, k-1/2), \quad E_\phi(i-1/2, k-1/2), \quad E_z(i-1/2, k), \\ & V_{er}(i, k-1/2), \quad V_{e\phi}(i-1/2, k-1/2), \quad V_{ez}(i-1/2, k), \\ & j_r(i, k-1/2), \quad j_\phi(i-1/2, k-1/2), \quad j_z(i-1/2, k), \\ & p_e(i-1/2, k-1/2), \quad n_e(i-1/2, k-1/2), \quad T_e(i-1/2, k-1/2). \end{aligned}$$

На каждом временном шаге (рис. 1) новые скорости ионов вычисляются по значениям электромагнитных полей в соответствующих точках [9], и этот шаг представляет собой переход от вычислений на эйлеровой сетке к вычислениям на лагранжевой сетке. Обратным переходом, от частиц к сетке, является вычисление ионной плотности и средних ионных скоростей.

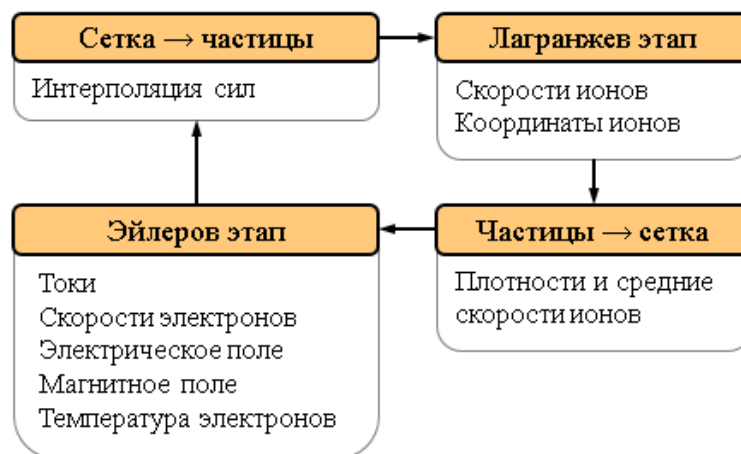


Рис. 1. Шаг цикла по времени

Fig. 1. Time cycle step

Большее количество модельных частиц в ячейке обеспечивает лучшую точность метода частиц-в-ячейках [10]. Гибридная модель предполагает, что, кроме ионов пучка, имеются ионы фона, так как плотность фона ненулевая.

Повышение числа ячеек при измельчении сетки ведет к увеличению затрат вследствие не только увеличения числа узлов, но и пропорционального увеличения полного числа частиц. Условная устойчивость численного алгоритма ограничивает величину временного шага

и увеличивает время расчетов. На практике для сетки 80×240 узлов потребовался временной шаг $2,5 \cdot 10^{-6}$. При этом теоретические оценки времени формирования стационарного пузыря составляют $10^4 \omega_{ci}^{-1}$, т. е. 10^4 в безразмерных единицах, и с таким временным шагом для решения потребуются длительные расчеты.

Распараллеливание является целесообразным путем ускорения вычислений. В нашем коде мы применяем смешанную декомпозицию, разделяя расчетную область вдоль оси z на подобласти и приписывая каждой подобласти группу процессов для обработки данных (рис. 2) [11]. Внутри каждой группы частицы распределяются почти равномерно: эту равномерность обеспечивает равномерное распределение частиц при инжекции и пересылке примерно одинакового количества частиц случайным процессам соседней группы. Для каждой группы определен главный процессор, отвечающий за вычисления на эйлеровом этапе (токи, скорости электронов, электрическое и магнитное поля, температура электронов), а также за рассылку данных по группе и сбор в группе двумерных массивов плотностей и средних скоростей. Все процессы обеспечивают вычисления на лагранжевом этапе и переход с сетки к частицам и обратно.

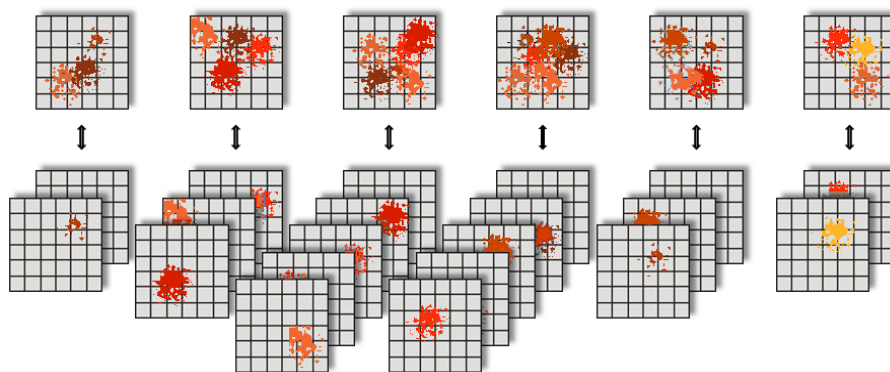


Рис. 2. Декомпозиция
Fig. 2. Decomposition

Такой алгоритм позволил сократить время расчета по сравнению с линейной декомпозицией (один процесс на группу) и ускорить в десятки раз вычисления по сравнению с однопроцессорным вариантом. В стандартных вычислениях центральные группы имеют максимальное количество процессов, с удалением от центра число процессов в группе линейно падает, и такая конфигурация является очень эффективной для моделирования начальной стадии эволюции системы.

Однако любое статическое распределение процессов по группам не может являться эффективной конфигурацией для нелинейно меняющегося распределения: распределение фоновых ионов в пространстве в начальный момент равномерно, но инжектируемые ионы вытесняют магнитное поле и фоновые ионы, размер диамагнитного «пузыря» растет, частицы двигаются к пробкам и могут отражаться от них и двигаться снова к центру ловушки, т. е. частицы переходят в соседние подобласти. Мы используем динамическую загрузку процессов, основываясь на среднем количестве частиц, обрабатываемых процессом в группе.

Другой вид распараллеливания – векторизация, когда одна инструкция применяется к нескольким значениям, приводя к высокоэффективным вычислениям. Использование явных схем позволяет применять автоматическую векторизацию циклов на эйлеровом этапе вычислений. Однако этот этап не занимает много времени, и его векторизация несущественно меняет время расчетов. Наиболее трудоемкими являются процедуры вычисления скоростей и координат частиц, а также вычисления плотностей и средних скоростей частиц [12].

Переход с эйлерова этапа на лагранжев обычно включает цикл по частицам для вычисления сил, действующих на частицу, которые вычисляются с помощью интерполяции в местоположение частицы по ближайшим узлам сетки (i, k) , $(i, k + 1)$, $(i + 1, k)$, $(i + 1, k + 1)$ (рис. 3, а).

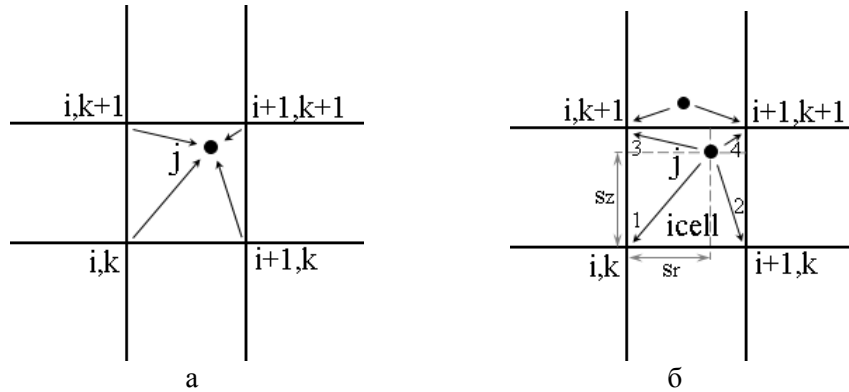


Рис. 3. Интерполяция сил (а), расчет плотности и плотности тока (б)
Fig. 3. Force interpolation (a), calculation of current density and density (b)

Зависимость от индексов (i, k) от номера частиц j ведет к косвенной индексации, и накладные расходы превышают полученный выигрыш от векторизации. Стандартное вычисление скоростей ионов и плотностей представляет собой цикл по частицам, вычисление номеров ближайших четырех узлов сетки, суммирование в этих узлах вкладов плотностей ионов и плотностей токов и последующее деление плотностей тока на плотности с учетом сдвинутых сеток [12]. Но в этом случае две частицы могут раздавать свои вклады в один и тот же узел сетки и вызывать зависимость по данным, и цикл не векторизуется. Оба случая требуют специальных алгоритмов во избежание описанных проблем [13; 14].

Обратим внимание на вычисление ионной плотности и плотностей токов (рис. 3, б). Стандартный алгоритм представляет собой цикл по частицам $j = 1..N_p$, где p – номер процесса. Например, для радиальной компоненты на каждом шаге цикла определяются минимальные индексы i, k ячейки, содержащей частицу j , расстояния до узлов ячейки $s_r, s_{z0,5}$ определяют веса заряда $q(j)$, поэтому величины плотности nr и радиальной компоненты плотности тока nvr пересчитываются следующим образом:

$$\begin{aligned} nr(i, k) &= nr(i, k) + (1 - s_r)(1 - s_z)q(j), \\ nr(i + 1, k) &= nr(i + 1, k) + s_r(1 - s_z)q(j), \\ nr(i, k + 1) &= nr(i, k + 1) + (1 - s_r)s_z q(j), \\ nr(i + 1, k + 1) &= nr(i + 1, k + 1) + s_r s_z q(j). \end{aligned}$$

Нижний индекс 0,5 у s_z напоминает о сдвинутой на полшага пространственной сетке в направлении z . Для векторизации мы используем алгоритм, предложенный в [15]. В этом случае вводятся двумерные массивы njr для плотности и nvr для плотности тока. Первый индекс массива определяет номер вершины ячейки m , второй индекс представляет собой номер ячейки $icell_r$. Частицы не могут раздавать свой заряд в разные ячейки, поэтому зависимость по данным отсутствует, и цикл может быть векторизован. Также частицы разделяются на блоки длины $lvect$ для того, чтобы блоки используемых данных вмещались в кэш, обеспечивая эффективные вычисления. При обработке каждого такого блока сначала в цикле $jloc = 1..lvect$ заполняются массивы длины $lvect$ значениями индексов $ic_r(jloc)$ ячейки (не узла), расстояний $s_r(jloc)$, $s_{z0,5}(jloc)$, зарядов $Q(jloc)$ и скоростей $v_r(jloc)$ частиц. Далее следует

расчет по частицам $j = 1..lvect$ вкладов плотностей и плотностей токов со вложенным автоматически векторизуемым циклом по вершинам m ячейки $icell_r = ic_r(jloc)$:

```
do m=1,4
  Δnr=( Ir(m)-sr)( Iz(m)-sz0.5) S(m) q(j)
  Δnvr=( Ir(m)-sr)( Iz(m)-sz0.5) S(m) q(j) vr(j)
  njr (m,icellr)= njr (m,icellr) + Δnr
  njvr (m,icellr)= njvr (m,icellr) + Δnvr
enddo
```

Здесь используются вектора $I_r = (1,0,1,0)$, $I_z = (1,1,0,0)$, $S = (1,-1,1-1)$, а массивы s_r , $s_{z0.5}$, Q , v_r имеют длину $lvect$. Пересчет значений $n_r(i, k)$ и $nv_r(i, k)$ в узлах сетки по величинам $n_j(m, icell)$ и $n_jv_r(m, icell)$ в вершинах ячеек требуют расчетов лишь на сетке и не занимают много времени, так как количество узлов сетки мало по сравнению с количеством частиц. Заметим, что алгоритм не требует сортировки частиц по ячейкам или вдоль оси.

Для вычислений компонент ϕ и z используются сдвинутые сетки, поэтому по аналогии вводятся массивы $n_{j\phi}$, $nv_{j\phi}$, n_{jz} , nv_{jz} и расстояния $s_{r0.5}$, s_z .

Результаты численных экспериментов

Для проведения численных экспериментов использовались значения магнитного поля $H_0 = 2$ кГ, плотности фоновой плазмы $n_0 = 10^{12} \text{ см}^{-3}$. Время измерялось в единицах $t_0 = 1/\omega_{ci} = 5.2 \cdot 10^{-8} \text{ с}$, $\omega_{ci} = eH_0/cm_i$ – ионная циклотронная частота. Единица длины $L_0 = c/\omega_{pi} = 22,8 \text{ см}$, $\omega_{pi} = \sqrt{4\pi n_0 e^2/m_i} = 1,3 \cdot 10^{10} \text{ с}^{-1}$ – ионная плазменная частота. Размеры области $R_{\max} = 4$, $Z_{\max} = 12$. Скорости измерялись в значениях альфвеновской скорости $V_A = \sqrt{4\pi m_i n_0} = 4.4 \cdot 10^8 \text{ с}$, средняя скорость инжектируемых ионов 0,5, тепловая скорость пучка 10^{-2} . На рис. 4, а приведены абсолютные значения магнитного поля в момент времени $t = 5$. В центральной области ловушки наблюдается каверна магнитного поля, т. е. область, где поле отсутствует или очень мало. Внутри радиуса $r = 1,5$ значения поля не превышают 1 % от H_0 . На рис. 4, б представлена плотность фоновых ионов в тот же момент времени $t = 5$. Высокая концентрация фоновых ионов достигается вне границы пузыря с высоким градиентом магнитного поля, граничный слой состоит из инжектированных ионов. Формирование диамагнитного пузыря согласуется с вытеснением фоновых ионов из центральной области и приводит к образованию области пониженной плотности фоновой плазмы.

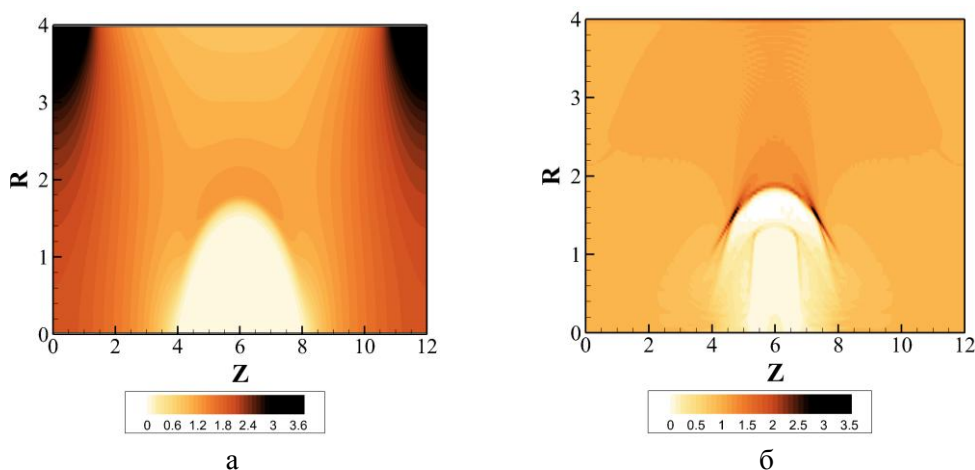


Рис. 4. Модуль магнитного поля (а), плотность ионов фона (б)
Fig. 4. Magnetic field module (а), background ion density (b)

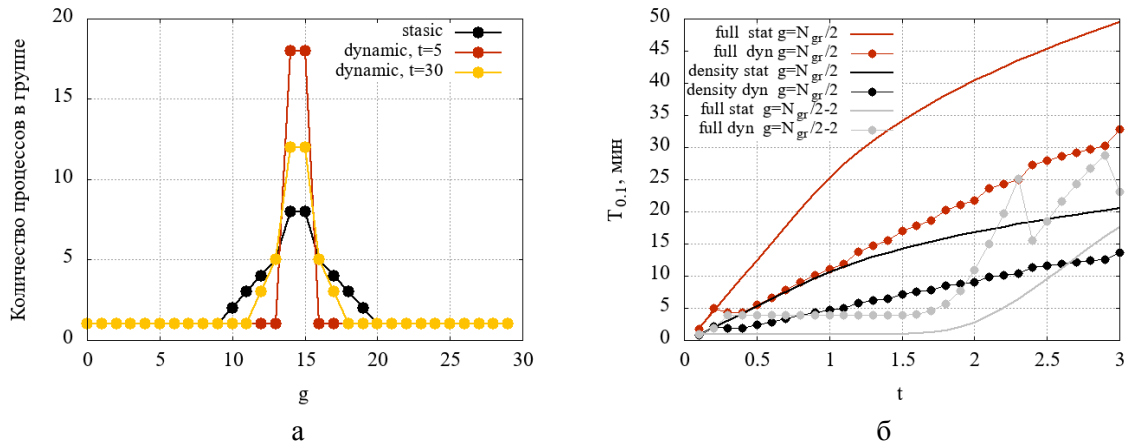


Рис. 5. Распределение процессов по группам (а), время работы частей программы (б)
 Fig. 5. Distribution of processes by groups (а), operating time of the program parts (b)

Расчеты, проведенные на сетке с количеством узлов 100×300 , потребовали $3 \cdot 10^6$ временных шагов для достижения момента $t = 3$ ($h_r = 4 \cdot 10^{-2}$, $h_z = 4 \cdot 10^{-2}$, $\tau = 10^{-6}$). В начальный момент времени в каждой ячейке находится 9 фоновых ионов (всего фоновых частиц $N_{bgr} = 2,7 \cdot 10^5$), на каждом шаге по времени вводится 1 частица пучка (при $t = 3$ полное число частиц $N = 3,27 \cdot 10^6$).

Тесты проводились на процессорах Intel Xeon Phi 7290 (Knights Landing, 1.5GHz), Intel Xeon E5-2697 v4 (Broadwell, 2.6 GHz) и Intel Xeon Platinum 8268 (Cascade Lake, 2.9 GHz). Код на языке Fortran с использованием MPI компилировался с помощью компилятора Intel версии 17.0.4 с ключом для векторизации `-xCORE-avx2`. Каждому процессу отвечает одно ядро процессора, количество процессов $np = 64$ было распределено на $N_{gr} = 30$ групп. На рис. 5, а приведено распределение процессов по группам для статической балансировки нагрузки и для динамической в моменты времени $t = 5$ и $t = 30$.

Для оценки времени работы алгоритмов использовалась величина $T_{0,1}$ времени исполнения процедурой или набором процедур 10^5 временных шагов $t^* = 10^5 \tau = 0,1$, так как она лучше, чем полное время работы, отражает влияние балансировки загрузки. На рис. 5, б приведены значения времени работы $T_{0,1}$ процедур в минутах. Сплошные линии отвечают версиям программы со статическим распределением процессов по группам, с точками – с динамическим, при этом балансировка происходит через время t^* . Красными линиями отмечены времена работы всей программы, черными – времена расчета плотности ионов и плотности тока в центральной группе (с номером $g = N_{gr}/2$), серыми – в группе с номерами $g = N_{gr}/2-2$. Линейный рост времени работы на начальном этапе (красная и черная линии) соответствует увеличивающемуся в центральных группах количеству частиц при инжекции, далее частицы покидают подобласть, и время растет медленнее. Соседние подобласти на начальном этапе содержат лишь фоновые частицы, частицы пучка появляются там позже (серые графики). Перераспределение процессов, например, в момент $t = 2.3$ позволяет снизить нагрузку на нецентральные группы, где частиц становится все больше, за счет распределения частиц на большее количество процессов группы. Время их работы в статической конфигурации меньше (серые графики), но общее время работы определяется максимальными временами работы по всем группам, и динамическая балансировка позволила уменьшить времена расчета на $\sim 30\%$. Для проведения расчетов до момента времени $t = 3$ потребовалось 25 часов в статической конфигурации и 15 часов при использовании динамической балансировки.

На рис. 6, а представлены времена расчета различными процессорами, красным обозначено время работы $T_{0,1}$ при $t = 3$ всего алгоритма, часть этого времени (желтого цвета) соответ-

ствуется времени расчета плотности и плотности тока. Три группы столбцов отвечают различным процессорам: первая соответствует процессорам Xeon Phi 7290 (KNL), вторая – Xeon E5-2697 (BDW), третья – Xeon Platinum 826 (CSL). Первый столбец в каждой группе отвечает исходному коду, второй – с векторизацией, третий – с динамической балансировкой, четвертый – с динамической балансировкой и векторизацией.

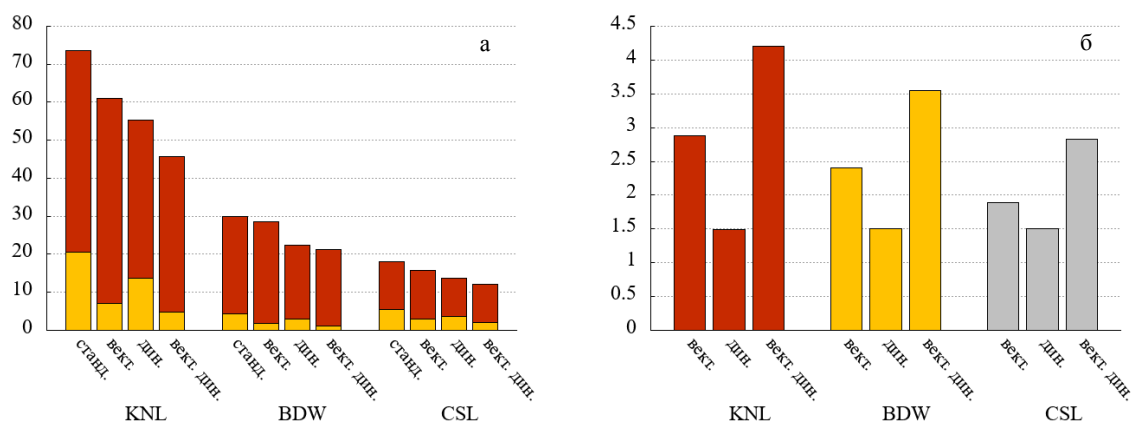


Рис. 6. Время работы версий кода с использованием различных процессоров (а), ускорение расчетов плотности ионов и плотности тока (б)

Fig. 6. Running time of code versions using different processors (a), acceleration of calculations of ion density and current density (b)

Наиболее медленно проводятся расчеты на процессорах KNL и быстрее всего на CSL, почти в три раза быстрее, чем KNL. Расчет плотностей без векторизации занимает примерно 30–40 % времени работы всех блоков, вычисления на сетке ~ 2 %, еще меньшее время занимают пересылки, а не векторизованный цикл вычисления координат и скоростей частиц занимает почти все оставшееся время. За счет этого достаточно большое ускорение в расчете плотности и плотности тока дает существенно меньшее снижение времени счета всей программы. На рис. 6, б показано ускорение процедуры расчета плотности ионов и плотности тока с применением векторизации (первый столбец), динамической балансировки (второй столбец) и векторизации вместе с динамической балансировкой (третий столбец) по сравнению с исходным вариантом кода. Первая группа отвечает вычислениям на процессорах KNL (красный цвет), вторая – BDW (желтый) и третья – CSL (серый). Векторизованный алгоритм расчета плотности и плотности тока позволяет снизить время вычислений в ~ 3 раза на KNL, ~ 2 на CSL и 1,5 на BDW, балансировка загрузки позволяет сократить соответствующие времена в 1,5 раза. Балансировка в совокупности с векторизацией позволила ускорить расчет плотностей в ~ 4 раза на процессорах KNL, ~ 3 на CSL и ~ 3,5 на BDW.

Процессоры KNL обеспечивают более эффективную векторизацию по сравнению с другими, но требуют большего времени расчетов, чем CSL с их менее эффективной векторизацией, но существенно более высокой тактовой частотой. В данной двумерной задаче цикл по вершинам ячейки имеет длину 4, в трехмерном случае вершин уже 8, и ускорение от векторизации должно быть существенно выше за счет длины вектора.

Расчеты проводились на кластере НКС-1П ССКЦ (ИВМиМГ СО РАН, Новосибирск).

Заключение

Представлен параллельный алгоритм моделирования динамики плазмы в диамагнитном режиме открытой магнитной ловушки. Алгоритм основан на гибридной модели с использо-

ванием метода частиц-в-ячейках. Распараллеливание проводится и по пространству, и по частицам. Результаты численных экспериментов демонстрируют ускорение расчетов для векторизованного алгоритма в 2–4 раза, векторизованного алгоритма с динамической балансировкой загрузки – в 3–4 раза. Процессоры KNL обеспечивают более эффективную векторизацию, меньшая эффективность векторизации с лихвой компенсируется высокой частотой процессоров CSL. Динамическое распределение процессов по областям с высокой плотностью частиц позволяет снизить время расчетов по сравнению с выбранной статической конфигурацией, оптимальной для определенного отрезка времени.

Список литературы / References

1. **Beklemishev A. D.** Diamagnetic “bubble” equilibria in linear traps. *Physics of Plasmas*, 2016, vol. 23, p. 082506.
2. **Ivanov A. A., Prikhodko V. V.** Gas dynamic trap: experimental results and future prospects. *UFN*, 2017, vol. 187, no. 5, p. 547–574.
3. **Bagryansky P. A., Akhmetov T. D., Chernoshtanov I. S., Deichuli P. P., Ivanov A. A., Lizunov A. A., Maximov V. V., Mishagin V. V., Murakhtin S. V., Pinzhenin E. I., Prikhodko V. V., Sorokin A. V., Oreshonok V. V.** Status of the experiment on magnetic field reversal at BINP. *AIP Conference Proceedings*, 2016, vol. 1771, p. 030015.
4. **Caprioli D., Pop A.-R., Spitkovsky A.** Simulations and Theory of Ion Injection at Non-relativistic Collisionless Shocks. *The Astrophysical Journal Letters*, 2015, vol. 798 (2), p. L28.
5. **Vshivkova L. V., Dudnikova G. I.** Hybrid Model of Particle Acceleration on a Shock Wave Front. In: Numerical Analysis and Its Applications: 6th International Conference, NAA 2016, Revised Selected Papers, Lecture Notes in Computer Science 10187, 2017, p. 737–743.
6. **Hockney R. W., Eastwood J. W.** Computer Simulation Using Particles. Bristol, Adam-Hilger, 1988.
7. **Braginskii S. I.** Transport Processes in a Plasma. *Reviews of Plasma Physics*, 1965, p. 205–311.
8. **Birdsall C. K., Langdon A. B.** Plasma physics via computer simulation. McGraw-Hill, 1985.
9. **Boris J. P.** Relativistic plasma simulation – optimization of a hybrid code. In: Proceedings of 4th Conference on Numerical Simulation of Plasmas. Washington DC, Naval Research Laboratory, 1970, p. 3–67.
10. **Lotov K. V., Timofeev I. V., Mesyats E. A., Snytnikov A. V., Vshivkov V. A.** Note on quantitatively correct simulations of the kinetic beam-plasma instability. *Physics of Plasmas*, 2015, vol. 22, p. 024502.
11. **Kireev S.** A Parallel 3D Code for Simulation of Self-gravitating Gas-Dust Systems. *Parallel Computing Technologies*, 2009, vol. 5698, p. 406–413.
12. **Boronina M. A., Chernykh I. G., Genrikh E. A., Vshivkov V. A.** Parallel Realization of the Hybrid Model Code for Numerical Simulation of Plasma Dynamics. *Journal of Physics: Conference Series*, 2019, vol. 1336, p. 012017.
13. **Nishiguchi A., Oritani S., Yabe T.** Vector calculation of particle code. *Journal of Computational Physics*, 1985, vol. 61, I.3, p. 519–522.
14. **Horowitz E. J.** Vectorizing the interpolation routines of particle-in-cell codes. *Journal of Computational Physics*, 1987, vol. 68, p. 56–65.
15. **Vincenti H., Lobet M., Lehe R., Sasanka R., Vay J.-L.** An efficient and portable SIMD algorithm for charge/current deposition in Particle-In-Cell codes. *Computer Physics Communications*, 2017, vol. 210, p. 145–154.

Материал поступил в редколлегию
Received
19.01.2021

Сведения об авторах

Боронина Марина Андреевна, кандидат физико-математических наук, научный сотрудник, Институт вычислительной математики и математической геофизики СО РАН (Новосибирск, Россия)

boronina@ssd.sccc.ru

ORCID 0000-0001-7090-4054

Вшивков Виталий Андреевич, доктор физико-математических наук, профессор, главный научный сотрудник, Институт вычислительной математики и математической геофизики СО РАН (Новосибирск, Россия)

vsh@ssd.sccc.ru

Киреев Сергей Евгеньевич, научный сотрудник, Институт вычислительной математики и математической геофизики СО РАН (Новосибирск, Россия), старший преподаватель, кафедра параллельных вычислений, факультет информационных технологий, Новосибирский государственный университет (Новосибирск, Россия)

kireev@ssd.sccc.ru

Information about the Authors

Marina A. Boronina, PHD, researcher, Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk, Russian Federation)

boronina@ssd.sccc.ru

ORCID 0000-0001-7090-4054

Vitaly A. Vshivkov, Professor, Head Researcher, Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk, Russian Federation)

vsh@ssd.sccc.ru

Sergey E. Kireev, Researcher, Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk, Russian Federation), Senior Lecturer, Department of Information Technologies, Novosibirsk State University (Novosibirsk, Russian Federation)

kireev@ssd.sccc.ru