

УДК: 519.6:533.9

Параллельные вычисления в дарвинской PIC-модели

Л. В. Бородачев^а, Д. О. Коломиец^б

Московский государственный университет им. М. В. Ломоносова, физический факультет,
Россия, 119991, ГСП-1, г. Москва, Ленинские горы, МГУ, д. 1

E-mail: ^а borodach2000@mail.ru, ^б kolomiets@darwincode.org

Получено 8 июля 2014 г.

Предлагается подход к параллельной реализации низкочастотных PIC-алгоритмов, учитывающий особенности безызлучательного (дарвинского) приближения электромагнитных полей разреженной плазмы. Обсуждаются его достоинства и специфика адаптации к основным типам программно-аппаратных платформ для высокопроизводительных вычислений

Ключевые слова: метод макрочастиц, модель Власова–Дарвина, PIC-алгоритм, параллельные вычисления, мультипроцессорные ЭВМ

Parallel calculations in the Darwin PIC-model

L. V. Borodachev, D. O. Kolomiets

Moscow State University, Faculty of Physics, MSU, GSP-1, Leninskiye Gory, Moscow, 119991, Russia

Abstract. — The approach to parallel implementation of low-frequency PIC-algorithms is proposed, taking into account peculiarity of the nonradiative (Darwin) field approximation. Its advantages and specifics of adaptation to the base computer types for high performance calculations are discussed.

Keywords: Particle Mesh method, Vlasov–Darwin model, PIC-algorithm, parallel calculations

Citation: *Computer Research and Modeling*, 2015, vol. 7, no. 1, pp. 61–69 (Russian).

Введение

Одним из наиболее эффективных методов численного анализа кинетики разреженной плазмы является метод макрочастиц (ММ) [Hockney, Eastwood, 1981]. Он основывается на предпосылке Власова, в рамках которой эволюция начального состояния любого ансамбля уединенных зарядов описывается с помощью фазовых траекторий сравнительно небольшого числа укрупненных (модельных) частиц, движущихся в самосогласованных (порождаемых ими) электромагнитных полях [Власов, 1950]. При этом степень достоверности такого описания в конечном итоге зависит от количества используемых макрочастиц, что обуславливает весьма жесткие (по сути, на грани возможного) требования к ресурсам ЭВМ — памяти и производительности [Полюдов, 1979]. Поэтому, несмотря на бурное развитие вычислительной техники в последние десятилетия, приведшее к качественным изменениям архитектуры и мощности компьютеров, полные электромагнитные постановки, по-прежнему, остаются уникальными: во-первых, общее количество требуемых в плазменных расчетах частиц растет как степень размерности конфигурационного пространства и, во-вторых, учет высокочастотных эффектов излучения приводит к исключительно мелкомасштабной дискретизации пространственно-временного континуума.

Отсюда понятен большой интерес к длинноволновым по характеру и несимметричным по фазовому пространству безызлучательным или дарвинским моделям [Darwin, 1920], адекватным весьма широкому кругу явлений плазмифизики, определяемых коллективными взаимодействиями.

Перспективы, открывающиеся в этом направлении, являются серьезным стимулом к построению дарвинских (безызлучательных) программных кодов, ориентированных на много-процессорные аппаратные платформы. Практической основой таких подходов служит технология параллельных вычислений [Воеводин, Воеводин, 2002], весьма естественная для метода макрочастиц и особенно легко реализуемая в PIC (Particle-In-Cell) модификации [Harlow, 1964], исходно содержащей в себе идею расщепления.

К сожалению, в настоящее время наиболее распространенные суперЭВМ (кластерного типа [Корнеев, 1999]), по сути, лишены средств рационального распараллеливания сложных программ (в частности, по методу макрочастиц) на уровне их трансляции. Сегодняшняя практика автоматизированного распараллеливания, без учета конкретной структуры вычислительного цикла и специфики обрабатываемых данных, как правило, выливается в организацию крайне неоптимальной сети межузловых коммуникаций, что сводит на нет эффективность параллельных вычислений в целом.

Для достижения приемлемой эффективности расчетов на современных мультипроцессорах с распределенной памятью требуются методики построения кодов с заданным параллелизмом и продуманной системой межузловых связей, определяемых особенностями реализуемых дискретных алгоритмов. Это положение особенно актуально для модели Власова–Дарвина, учитывая специфику ее наиболее сложной неявной численной аппроксимации по методу макрочастиц [Бородачев, 1991; Бородачев, 2005].

В настоящей работе авторы предлагают эффективный подход к распараллеливанию безызлучательных (дарвинских) PIC-алгоритмов и обсуждают специфику его реализации применительно к основным типам программно-аппаратных платформ для высокопроизводительных вычислений.

Параллельные PIC-алгоритмы

Идея использования параллельных вычислений в практической реализации моделей частиц высказывалась уже достаточно давно, в частности, одним из авторов настоящей работы [Бородачев, 1990]. Однако методическое оформление она получила в девяностые годы, когда

появились реально действующие массивно-параллельные системы с соответствующим программным обеспечением [Корнеев, 1999]. При этом в контексте общей оптимизации счета базовый вопрос уже исходно состоял в том, каким образом распределять данные и вычисления между процессорными элементами (узлами) таких систем. (Здесь и далее под данными мы понимаем атрибуты частиц (заряд, массу, фазовые координаты), сеточные значения источников (плотностей заряда и тока) и компонент поля (магнитного и электрического)). Его решение, очевидно, требует анализа различных по характеру и степени влияния на производительность вычислений факторов, например архитектуры мультикомпьютера, эффективности коммуникационной среды, возможностей мониторинга задачи операционной системой и тому подобное. Оставляя в стороне «программно-аппаратные» факторы, которые должны учитываться конкретной вычислительной платформой, остановимся на моментах, определяемых модельными особенностями реализуемых PIC-алгоритмов. И в этом контексте прежде всего рассмотрим общепринятый подход к распараллеливанию дискретных электромагнитных моделей плазмы на базе метода декомпозиции [Walker, 1992], имея в виду его приложение к безызлучательным кодам.

Метод декомпозиции

В основе настоящего метода, имеющего различные (как правило, определяемые компромиссом между счетной и коммуникационной загрузкой процессорных элементов) модификации — декомпозиция по Эйлеру, иерархическая декомпозиция, декомпозиция для архитектуры гиперкуб и так далее — лежит общий принцип разделения вычислительной области на подобласти (домены), которые вместе с принадлежащими им частицами распределяются по отдельным вычислительным узлам. Поэтому, не вдаваясь в детали различных модификаций, обсудим характерные черты этого подхода в указанном выше аспекте.

Основным преимуществом метода декомпозиции является возможность решения задач с высоким пространственным разрешением, в которых число узлов сетки настолько велико, что не укладывается в объем оперативной памяти отдельного вычислительного узла. Эта ситуация типична для полных самосогласованных моделей плазмы, где максвелловское описание полевой части предполагает априорное существование коротковолновых мод электромагнитного поля в системе.

Однако ряд трудностей, обусловленных самой концепцией метода декомпозиции, ставит под сомнение его эффективность при параллельной реализации низкочастотных, в частности дарвинских, моделей плазмы. Остановимся на основных, с нашей точки зрения, моментах.

Прежде всего, переход любой частиц из одной подобласти пространства в другую влечет за собой необходимость пересылки ее атрибутов (координат и скорости) из одного процессорного элемента в другой, причем объемы этих пересылок растут с увеличением соотношения количества граничных узлов всех подобластей к числу их внутренних узлов. Таким образом, при увеличении количества вычислительных узлов существенно возрастает и объем межпроцессорных коммуникаций, что естественно ухудшает как масштабируемость кода, так и его экономичность. Более того, объем пересылок зависит не только от шага интегрирования по времени, что вполне естественно, но и от характеристик модельной плазмы: температуры и скорости направленного переноса частиц. Таким образом, численная эффективность становится функцией не только дискретных, но и физических параметров системы, что вряд ли можно отнести к достоинствам метода.

В дополнение к этому отдельной проблемой становится решение (особенно при его итерационном характере [Бородачев, Коломиец, Литвинюк, 2006]) полевых уравнений, требующее теперь либо интенсивных межпроцессорных обменов сеточными значениями, либо организации внутреннего итерационного процесса с обменом граничными сеточными значениями соседних подобластей [Rice, Vavalis, Yang, 1997; Braverman, Israeli, Averbuch, 2005]. Помимо значительного увеличения и без того немалых накладных коммуникационных расходов, последнее

актуализирует вопросы сходимости глобального сеточного решения. Очевидно, что и то и другое резко снижает эффективность полевого блока и, как следствие, кода в целом.

Наконец, существенным недостатком метода декомпозиции является априорная разбалансировка вычислительной нагрузки с течением времени. Действительно, в процессе перехода частиц из одной подобласти в другую достаточно часто возникает ситуация, при которой одному или нескольким вычислительным модулям приходится обрабатывать существенно больше частиц, чем остальным (например, при образовании пространственно локализованной неоднородности в плотности частиц). В этом случае значительная часть «недогруженных» модулей, закончив работу, будет простаивать; в результате общая производительность системы резко снижается. Более того, миграция частиц в какую-либо одну подобласть может привести к исчерпанию свободной оперативной памяти соответствующего вычислительного узла, что, в свою очередь, приведет к аварийному завершению всего процесса. Чтобы избежать подобных проблем, приходится прибегать к методам адаптивной декомпозиции, в которых предусматривается возможность динамического изменения конфигурации подобластей, построения специальных карт пересылок частиц и т. п. Однако применение таких методов неизбежно приводит к весьма значительному (зачастую на грани разумного) усложнению кода [Андрианов, Ефимкин, 2009].

Вместе с тем безызлучательное полевое представление предполагает заведомое обрезание коротковолновой части спектра электромагнитных волн, присутствующих в системе. При этом естественное для него выделение продольных и поперечных компонент в векторных величинах полей и токов на основе разложения Гельмгольца делает органичными дробно-мерные постановки с редуциацией конфигурационного пространства [Бородачев, 1993]. Очевидным следствием этих положений в контексте настоящей работы является возможность использования относительно грубых пространственных сеток, число узлов которых (на измерение) может быть при необходимости существенно повышено за счет перехода к несимметричной по фазовой геометрии формулировке задачи без потери ее физической адекватности.

В этой связи представляется разумным применение альтернативного подхода к распараллеливанию PIC-алгоритмов, учитывающих особенности самосогласованных дарвинских моделей плазмы.

Метод разделения частиц

Этот подход в определенном смысле является развитием методики реализации низкочастотных PIC-алгоритмов на ЭВМ с малым объемом оперативной памяти, успешно применяемой одним из авторов еще в конце 70-х годов прошлого века в ИПМ им. М. В. Келдыша РАН [Бородачев, Сигов, 1979]. Суть ее состояла в совмещении парциальной (пакетной) обработки частиц, хранящихся на внешнем носителе, с динамической подкачкой необходимых в процессе счета программных блоков (в частности, модуля расчета самосогласованных полей) в ОЗУ через специальные буферные зоны — распараллеливание данных и вычислений на базе одного процессора.

В предлагаемом методе все частицы дискретной плазменной модели равными долями распределяются по вычислительным узлам вне зависимости от их пространственной локализации, при этом каждый такой узел обладает собственной копией сеточных значений полей. По завершении очередного шага временной эволюции самосогласованной системы все вычислительные узлы суммируют вклады своих частиц в единые массивы сеточных значений источников (плотность заряда, плотность тока и другие необходимые моменты функции распределения частиц), которые, в свою очередь, передаются на выделенный узел для решения полевых уравнений.

Преимущество данной методики распараллеливания для низкочастотных (дарвинских) алгоритмов основывается на следующих положениях.

Во-первых, в типичном компьютерном эксперименте на базе PIC-метода бóльшая часть вычислительных затрат (порядка 90 %) приходится именно на продвижение частиц и получение сеточных источников, а не на вычисление полей. Таким образом, эффективное распараллеливание лишь этой части вычислений позволяет добиться существенного ускорения счета в целом. При этом достигается наиболее равномерное распределение данных и вычислений по узлам, связанным с обработкой частиц, так что актуальная для методов на основе сегментации области проблема разбалансировки вычислительной нагрузки здесь, по сути, не стоит.

Во-вторых, программная реализация метода разделения частиц намного проще по сравнению с методом декомпозиции области, где существуют указанные выше алгоритмические проблемы, связанные как с оптимизацией процедуры пересылки частиц, так и с получением корректного решения полевых уравнений, особенно в дробно-мерных постановках. При этом структура межпроцессорных взаимодействий по методу разделения частиц допускает относительно несложную интеграцию в уже существующие последовательные коды, которые легко адаптируются к мультипроцессорным ЭВМ как с распределенной, так и с общей памятью.

В-третьих, высокая эффективность параллельного кода в случае не слишком больших сеток и большого числа частиц (то есть когда численная постановка задачи позволяет иметь копию всех необходимых сеточных значений на каждом вычислительном узле, а количество частиц, приходящихся на долю каждого процессора, достаточно велико, чтобы работа по ним составляла бóльшую часть времени счета), типично именно для дискретных безызлучательных моделей плазмы. Более того, эффективность резко возрастает с увеличением отношения числа частиц к числу сеточных узлов, ибо повышается не только экономичность расчета (что очевидно), но и его достоверность в силу значительного улучшения физических свойств (бесстолкновительности, идеальности и стохастичности) модельной системы [Hockney, Eastwood, 1981].

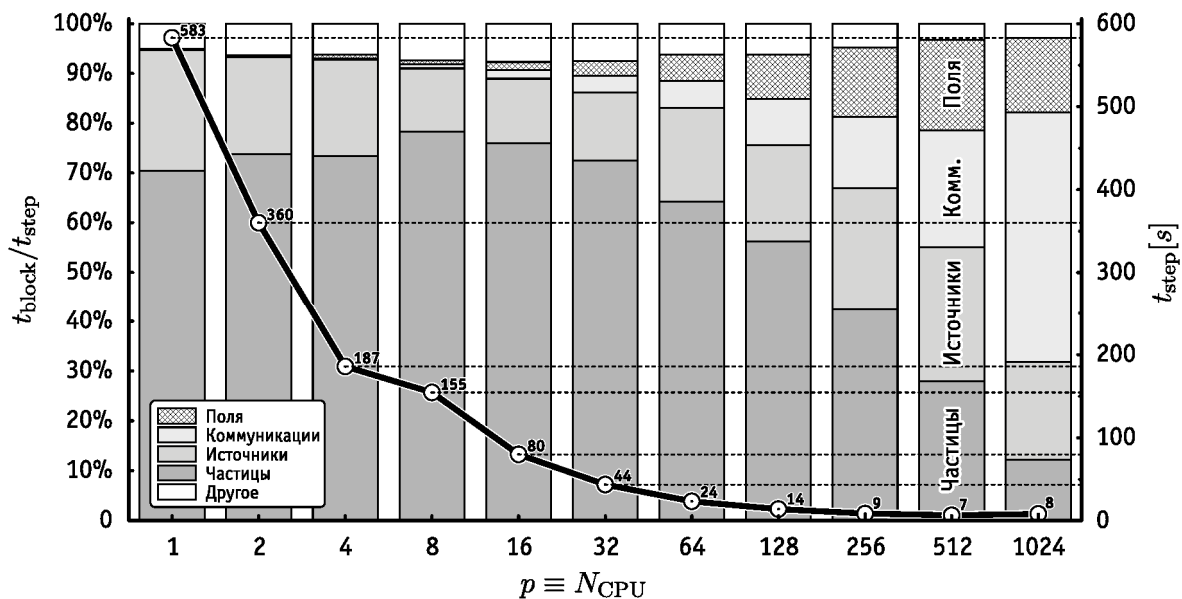


Рис. 1. Соотношение времен выполнения различных стадий вычислительного процесса t_{block} / t_{step} и суммарное время одного шага t_{step} (сплошная линия, масштабная шкала справа) в зависимости от числа процессоров (ядер)

В-четвертых, коммуникационные издержки данного метода, в отличие от метода декомпозиции области, не зависят ни от количества модельных частиц, ни от их температуры и потоковой скорости, а являются лишь функцией размера сетки. При этом в случае правильно организованных потоках выполнения («нитях») временные затраты на межпроцессорные коммуникации, по нашим оценкам (см. рис. 1), достаточно слабо коррелируют и с количеством вычислительных узлов (конкретнее, на типичном вычислительном кластере с MPI-технологией

время пересылок (сбора парциальных сеточных вкладов и раздачи сеточных значений полей) зависит от количества вычислительных узлов примерно как $\log_2(N_{\text{CPU}})$, что согласуется с положениями работы [Антонов, 2004]. Заметим попутно: указанные обстоятельства делают более предсказуемыми и оценки общего времени счета конкретной задачи, что является дополнительным плюсом.

Вместе с тем подход не лишен и определенных ограничений. Прежде всего, наличие одного выделенного (корневого) узла для решения полевых уравнений ограничивает масштабируемость кода. Действительно, при любом количестве вычислительных модулей, участвующих в параллельном счете, общее время вычислений физически не может быть меньше суммарного времени расчета полей (которое не зависит от количества участвующих процессоров) и копирования их сеточных значений (которое увеличивается с увеличением числа участвующих модулей). Таким образом, в принципе возможна ситуация, когда при дальнейшем увеличении количества вычислительных узлов время счета начнет увеличиваться. Помимо этого, как было отмечено выше, каждый участвующий в счете узел снабжается полной копией сеточных полей, что, вообще говоря, ограничивает размер модельной области объемом локальной оперативной памяти используемого кластера. (В более легкой форме эта проблема присутствует и в методе декомпозиции: попытка решения полной электромагнитной задачи большого пространственного масштаба, предполагающая значительную сегментацию области, может привести к резкому доминированию коммуникационных затрат в общем объеме вычислений и, как следствие, крайне низкой рентабельности такого расчета.)

Вместе с тем использование предлагаемого подхода в 2.5-мерном параллельном коде **DarWin** [Бородачев, Коломиец, 2010] и практика его успешной эксплуатации на кластере СКИФ МГУ «Чейбышев» [Бородачев, Коломиец, 2010; Borodachev, Kolomiets, 2011] с различным количеством выделяемых процессорных ядер позволяют говорить, что указанные ограничения являются сравнительно слабыми.

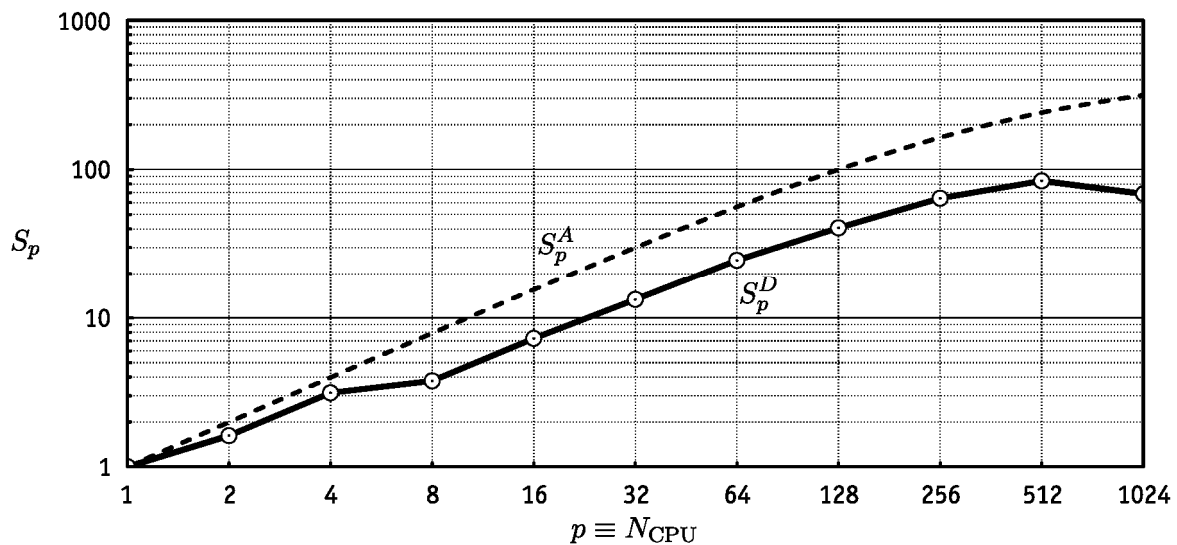


Рис. 2. Зависимость ускорения вычислений от числа процессоров (ядер). S_p^D — фактическое ускорение кода **DarWin**, S_p^A — теоретическое ускорение по закону Амдала

Действительно, несложные расчеты показывают, что для типового элемента вычислительного кластера с объемом памяти 8 гигабайт предельное количество сеточных узлов (учитывая, что хотя бы половина памяти занята частицами) составляет величину порядка 10^7 , то есть в случае 2.5-мерной фазовой геометрии предел по размеру сетки составляет 4096 узлов на пространственное измерение. Величина более чем достаточная при исследованиях в интересующей нас области низкочастотной плазмифики.

Далее, график фактического ускорения вычислений (рис. 2), полученный в ходе исследований масштабируемости того же кода **DarWin** на модельной задаче об эволюции двухкомпонентной однозарядной плазмы с анизотропным распределением тепловых скоростей («вайбелевский» тест: $N_{\text{part}} \approx 2.5 \cdot 10^9$, $N_{\text{cell}} = 512 \times 512$), которая решалась в тестовом режиме с использованием различного числа доступных процессорных ядер (от 1 до 1024) кластера, показывает, что критическая точка масштабируемости соответствует 512 процессорам, что составляет величину, достаточно большую для реального счета в стандартно используемом пакетном режиме обработки задач. Отметим при этом, что линейный по сути характер реальной кривой ускорения, практически совпадающей с кривой теоретического прогноза по закону Амдала [Воеводин, Воеводин, 2002], является безусловным практическим подтверждением эффективности предлагаемой методики распараллеливания безызлучательных PIC-алгоритмов (локальное расхождение указанных кривых объясняется использованием кластера с узлами, состоящими из четырех процессоров, попарно коммутируемых на общую память вычислительного модуля).

Специфика реализации метода разделения частиц

Представляется разумным дать некоторые рекомендации по реализации метода на мультипроцессорных системах основных типов. Для координации вычислительных процессов в массивно-параллельных компьютерах, по-видимому, оптимален интерфейс обмена сообщениями MPI (Message Parsing Interface) [Антонов, 2004]. Поскольку MPI, реализованный в виде сторонних библиотек, является стандартом де-факто в современных высокопроизводительных вычислениях, его применение обеспечивает легкую переносимость кода на основные типы программно-аппаратных платформ, работающих под управлением различных операционных систем (в частности, наиболее распространенных ОС Linux и Unix).

Вследствие того, что концепция метода разделения частиц весьма практична с точки зрения единообразия вычислений, производимых всеми процессорами, схема обменов оказывается также весьма простой. Фактически обмены ограничиваются только коллективными операциями: суммирование парциальных сеточных плотностей, полученных каждым из вычислительных элементов, с передачей результата в корневой модуль и последующая широковещательная рассылка найденных сеточных полей с него на все остальные процессоры.

Для SMP-кластеров, состоящих из многопроцессорных узлов, возможны две стратегии: распараллеливание с использованием только лишь MPI и гибридный подход, при котором в пределах одного вычислительного узла используется стандарт OpenMP (Open Multi-Processing) [Chandra et al., 2000], организующий обмены между процессорами через общую память, а межузловые коммуникации осуществляются посредством MPI.

Выбор того или иного подхода зависит от эффективности конкретных реализаций MPI и OpenMP, применяемых на вычислительном комплексе. Если библиотека MPI, используемая в работе ЭВМ, способна учитывать тот факт, что некоторые процессы выполняющейся параллельной задачи работают на одном и том же мультипроцессоре, и организует для них быстрый обмен данными без использования общей коммуникационной сети, тогда оба этих подхода обеспечивают сравнимую производительность. В остальных случаях гибридный подход обычно оказывается более эффективным, так как часть сравнительно медленных MPI-пересылок заменяется более быстрыми обменами через общую память посредством OpenMP.

Отдельно отметим специфику метода для суперЭВМ с общей памятью. Здесь реализация параллельного счета существенно упрощается, поскольку сеточные значения доступны любому из процессоров. Остается лишь разделить работу по продвижению частиц и сбору источников. Существует, однако, ряд особенностей, способных повлиять на общую производительность вычислений. В частности, большая вероятность попытки одновременной записи множеством процессоров сеточных вкладов своих частиц в массив источников, что, вообще говоря, дает не-

предсказуемый результат. Чтобы избежать этой проблемы, процедура записи должна быть помечена как критическая секция, что создает ее очередность по процессорам. При этом суммарное время последовательных записей может оказаться соизмеримым с временем вычислений. Ситуация усугубляется тем, что сами процедуры входа в критическую секцию и синхронизации процессоров для поочередной записи также весьма затратны.

Для решения проблемы критической секции можно проводить расчет источников, используя для каждого процессора временные массивы, которые далее суммировать в единый, причем параллельно по непересекающимся между процессорами множествам индексов массива. Заметим, что эта методика практически напрямую возвращает нас к базовому варианту метода, описанному выше. Указанный прием, как показывает практика, позволяет существенно поднять эффективность параллельных вычислений, однако требует определенного расхода памяти на временные массивы.

Заключение

Сравнительный анализ рассмотренных методик распараллеливания вычислений в области безызлучательного PIC-моделирования позволяет сделать следующий вывод. Применение метода декомпозиции области, по-видимому, оправдано лишь для полностью трехмерных кодов, причем в случае, когда размеры пространственных сеток существенно превышают возможности отдельного вычислительного узла с точки зрения доступного объема локальной памяти. В дробномерных постановках, свойственных дискретному дарвинскому моделированию, сложность программной реализации алгоритмов в совокупности с проблемами получения глобального сеточного решения полевых уравнений и минимизации коммуникационных затрат (резко возрастающих при множественной сегментации расчетной области) делают метод разделения частиц более предпочтительным. При этом низкочастотный характер процессов, свойственных безызлучательному моделированию, обуславливают, как указывалось выше, адекватность достаточно крупномасштабной дискретизации задач вычисления, что позволяет использовать в расчетной области сравнительно небольшие пространственные сетки, оптимальные как для хранения в локальной памяти вычислительного узла кластера, так и для создания временных массивов в общей памяти мультипроцессорных ЭВМ.

Список литературы

- Андреанов А. Н., Ефимкин К. Н.* Подход к параллельной реализации метода частиц в ячейках // Препринт ИПМ им. М. В. Келдыша. — М., 2009. — № 9. — 20 с.
- Антонов А. С.* Параллельное программирование с использованием технологии MPI. — М.: Изд-во МГУ, 2004. — 71 с.
- Бородачев Л. В.* Неявная аппроксимация уравнений движения дарвинской модели плазмы // ЖВМ и МФ. — 1991. — Т. 31, № 6. — С. 934–939.
- Бородачев Л. В.* Многомерные алгоритмы физики плазмы и параллельные вычисления // Автоматизация создания мат. обеспечения и архитектуры систем реального времени. Мат. всесоюзной школы. — Иркутск, 1990. — С. 189–201.
- Бородачев Л. В.* Численная интерпретация полевого описания в дискретной дарвинской модели с неявной схемой расчета динамики частиц // Мат. моделирование. — 2005. — Т. 17, № 9. — С. 53–59.
- Бородачев Л. В.* К проблеме математического моделирования безызлучательной плазмы // Вестник МГУ. Сер. 3. — 1993. — С. 87.
- Бородачев Л. В., Коломиец Д. О.* Расчет динамики частиц в безызлучательной модели плазмы // Мат. моделирование. — 2010. — Т. 22, № 10. — С. 83–92.

- Бородачев Л. В., Коломиец Д. О.* Электронная вайбелевская неустойчивость плазмы с температурной анизотропией. Вестник МГУ. Сер. 3. — 2010. — № 2. — С. 14–18.
- Бородачев Л. В., Коломиец Д. О., Литвинюк В. В.* Численное решение уравнений для соленоидального электрического поля в дарвинской модели плазмы // Вестник МГУ. Сер. 3. — 2006. — С. 14–17.
- Бородачев Л. В., Сигов Ю. С.* Численные эксперименты по параметрическому возбуждению магнитоактивной плазмы // Препринт ИПМ им. М. В. Келдыша. — М., 1979. — № 65. — 31 с.
- Власов А. А.* Теория многих частиц. — М.–Л.: ГИТТЛ, 1950. — 348 с.
- Воеводин В. В., Воеводин Вл. В.* Параллельные вычисления. — СПб.: БХВ-Петербург, 2002. — 608 с.
- Корнеев В. В.* Параллельные вычислительные системы. — М.: Нолидж, 1999. — 320 с.
- Полюдов А. Н.* Реализация моделей макрочастиц на ЭВМ // Препринт ИПМ им. М. В. Келдыша. — М., 1979. — № 160. — 32 с.
- Borodachev L. V., Kolomiets D. O.* Single-Species Weibel Instability of Radiationless Plasma // J. Plasma Phys. — 2011. — Vol. 77. — P. 277–288.
- Braverman E., Israeli M., Averbuch A. A.* Hierarchical 3-D Direct Helmholtz Solver by Domain Decomposition and Modified Fourier Method // J. Sci. Comput. — 2005. — Vol. 26, No. 5. — P. 1504–1524.
- Chandra R., Menon R., Dagum L., Kohr D., Maydan D., McDonald J.* Parallel Programming in OpenMP. — New York: Morgan Kaufmann, 2000. — 231 p.
- Darwin C. G.* Dynamical Motions of Charged Particles // Phil. Mag. — 1920. — Vol. 39. — P. 537–551.
- Harlow F. H.* The Particle-in-Cell Computing Method in Fluid Dynamics // Methods Comput. Phys. Edited by Alder B. Fernbach S., Rotenberg M. — New York: Acad. Press, 1964. — Vol. 3. — P. 319–343.
- Hockney R. W., Eastwood J. W.* Computer Simulation Using Particles. — New York: McGraw-Hill, 1981. — 540 p.
- Rice J. R., Vavalis E. A., Yang D.* Analysis of a nonoverlapping domain decomposition method for elliptic partial differential equations // J. Comput. and Appl. Math. — 1997. — Vol. 87. — P. 11–19.
- Walker D. W.* The Hierarchical Spatial Decomposition of Three Dimensional Particle-in-Cell Plasma Simulations on MIMD Distributed Memory Multiprocessors // Oak Ridge National Laboratory, report ORNL/TM-12071. — 1992. — 20 p.