

can be behind car in the blind spot. The point is in avoiding collision of the car with any obstacle (other car, building, person, animal, etc.). There are two parameters to be considered: driving speed and direction. If the distance between one car and other car or any obstacle gets smaller, a vehicle can change either driving speed or direction, thus ensuring safe road traffic. This paper considers vehicle movement through a T-junction as an example and shows that applying the considered generalized algorithm allows ensuing safe road traffic.

Key words: weakly formalized information; algorithm; information processing; road traffic safety.

About the author: Anton Alexandrovich Kopyltsov, Senior Lecturer at the Department of Automated Information Processing Systems, Saint-Petersburg State Electrotechnical University «LETI».

Place of employment: Saint-Petersburg State Electrotechnical University «LETI».

УДК 51:004.942

А.С. Трунов, Л.И. Воронова, В.И. Воронов
Москва, Россия

РАЗРАБОТКА ПАРАЛЛЕЛЬНОГО АЛГОРИТМА ДЛЯ ИНФОРМАЦИОННО-ИССЛЕДОВАТЕЛЬСКОЙ СИСТЕМЫ «MD-SLAG-MELT» НА ОСНОВЕ ТЕХНОЛОГИИ CUDA

Аннотация. Одним из приоритетных направлений современной науки является создание новых материалов с заранее заданными свойствами. В этой области широко применяется компьютерное моделирование (КМ), в том числе метод молекулярной динамики, позволяющий определять целый комплекс свойств (структурные, термодинамические, транспортные) и исследовать взаимосвязи наноструктуры и физико-химических свойств.

Для проведения КМ создаются автоматизированные информационные системы (АИС), главной целью которых является расширение границы исследований, оптимизация научной работы и ускорение проведения исследований. Одной из таких систем является ИИС «Шлаковые расплавы» (ИИС «MD-SLAG-MELT») [1].

Особенности предметной области ИИС таковы, что в настоящее время без применения методов распределения вычислений удастся просчитывать за приемлемое время групповое поведение систем, содержащих в лучшем случае десятки тысяч частиц. Однако существует ряд задач, в частности, связанных с определением пространственных наноразмерных неоднородностей, для которых необходимо увеличение размерности модельной системы до миллионов частиц. Промоделировать подобную систему на локальном компьютере в последовательном режиме практически не удастся, т.к. это связано с большими временными затратами, эксперимент может занять несколько месяцев. Решение подобных задач требует использования распределенных вычислений.

В статье описан разработанный авторами параллельный алгоритм, адаптирующий существующий (legacy application) линейный алгоритм расчета сил межмолекулярного взаимодействия под распределенные вычисления на центральном и графическом процессорах вычислительного устройства и имплементированный в программную среду ИИС «MD-SLAG-MELT». В основе разработанного распределенного алгоритма лежит технология параллельного программирования CUDA.

Ключевые слова: Информационно-исследовательская система (ИИС) «MD-SLAG-MELT»; молекулярная динамика; распределенные вычисления; параллельный алгоритм; CUDA

Сведения об авторах: Трунов Артем Сергеевич¹, старший преподаватель кафедры информационных систем и моделирования РГГУ; Воронова Лилия Ивановна², заведующая кафедрой информационных систем и моделирования РГГУ, профессор кафедры математической кибернетики и информационных технологий МТУСИ; Воронов Вячеслав Игоревич³, доцент кафедры информационных систем и моделирования РГГУ.

Место работы: Российский государственный гуманитарный университет (ФГОБУ ВПО «РГГУ») ^{1,2,3}; Московский технический университет связи и информатики (ФГОБУ ВПО «МТУСИ») ².

Контактная информация: 111672, г. Москва, ул. Новокосинская, д. 40, кв. 204; тел: 8-916-068-68-38. E-mail voronova2001@mail.ru.

Информационно-исследовательская система (ИИС) MD-SLAG-MELT [1; 3; 7] – программный комплекс, реализующий компьютерные эксперименты по исследованию физико-химических свойств и структуры материалов, ресурс по направлению компьютерного материаловедения.

Базовым методом в ИИС является метод молекулярной динамики (МД), который включает анализ сложного силового взаимодействия между частицами модельной системы. Размер системы (число исследуемых частиц) при МД-моделировании является критически важным. С увеличением числа частиц до 10^6 линейные размеры модельного куба достигают нанометров, что позволяет получать результаты нового качества, обладающие практической значимостью. Однако расчет систем, содержащих 10^5 – 10^7 частиц, требует серьезных временных затрат и вычислительных ресурсов, что делает невозможным проведение компьютерных экспериментов без привлечения высокопроизводительных вычислений.

В условиях ограниченных аппаратных ресурсов повышение производительности вычислений возможно за счет привлечение графического процессора видеокарты, используемого для компьютерных экспериментов вычислительного устройства (персонального компьютера). Делегирование части вычислений исходной задачи графическому процессору позволит разгрузить центральный процессор, что сократит общее время обработки. Для реализации описанной модели (модель распределенных вычислений MapReduce) необходимо разработать и имплементировать параллельный алгоритм, адаптирующий существующий линейный алгоритм расчета сил межмолекулярного взаимодействия под распределенные вычисления на центральном и графическом процессорах вычислительного устройства.

В статье описан разработанный авторами и реализованный в ИИС MD-MELT параллельный алгоритм расчета сил меж-

молекулярного взаимодействия. В основе разработанного распределенного алгоритма лежит технология параллельного программирования CUDA (Compute Unified Device Architecture).

Разработка параллельного алгоритма для вычисления результирующих сил взаимодействия между всеми частицами состоит из нескольких этапов: декомпозиция вычислений на независимые подзадачи, выделение информационных зависимостей между подзадачами, масштабирование подзадач – определение необходимой для решения вычислительной схемы и распределение подзадач между исполнительными процессорами [6].

Модель предметной области – коррелированная система N-частиц. Под коррелированной системой понимается система взаимодействующих объектов (частиц), в которой часть характеристик индивидуального объекта или системы в целом зависит от совокупности характеристик всех остальных объектов. Каждая модельная частица имеет набор сохраняющихся и переменных атрибутов. В этом случае МД-моделирование представляет собой численное решение краевой задачи Коши.

Для *распределенного* МД-моделирования коррелированной системы N-частиц авторами разработана модель неоднородных дескрипторов. На основе концептуальной модели МД-метода и тщательного анализа программного кода legacy application – локального МД-приложения построены наборы дескрипторов, описывающих бизнес-логику МД-приложения с точки зрения разделения его кода на блоки, пригодные для распределения.

Выделено два класса – одночастичные дескрипторы ($D1s(i)$, $D1v(i)$) и агрегаторы (двух- и трехчастичные ($D\Sigma 2(i)$, $D\Sigma 3(i)$) дескрипторы). Оба класса предполагают возможность параллельного расчета дескрипторов на разных вычислителях. Однако, если одночастичные дескрипторы можно произвольно распределять по вычислителям, то агрегаторы (содержат эле-

менты, описывающие перекрестные отношения разных порядков между одночастичными дескрипторами и/или агрегаторами) являются «зависимыми» от одночастичного дескриптора и рассчитываются на том же вычислительном устройстве, что и «родительский» одночастичный дескриптор. Предложенный подход позволяет отвлечься от конкретного наполнения элементов дескрипторов, перенести акцент с описания физических взаимодействий в системе на информационное описание перераспределения потоков данных между дескрипторами [2; 4].

Метод равномерной загрузки вычислителей

Равномерное распределение задач между исполнительными процессорами повышает общую эффективность вычислений (балансировка системы за счет отсутствия простоев). При этом необходимо учитывать загрузку нитей (единичных процессов, выполняемых на единичных процессорах) – обеспечить не только минимизацию их простоев, но и сокращение количества операций по передаче данных от центрального процессора графическому при вычислениях.

Наиболее трудоемкая операция на каждом шаге моделирования – расчет взаимодействий между объектами. Она требует выполнения числа операций, квадратичного по отношению к числу объектов. Для реализации параллельного расчета нужно разделить систему из N объектов на блоки. Каждый блок рассчитывается на отдельном процессоре.

Предварительная схема разбиения задачи на подзадачи: каждый процесс выполняет задачу по подсчету взаимодействия между двумя выбранными частицами системы. Внутри блока (группы программно организованных процессоров) выполняется задача по агрегации результатов выполнения на более низком уровне. Таким образом, результат работы каждого блока – подсчет взаимодействия всех частиц с дан-

ной. Верхний уровень системы (сетка – совокупность программно организованных блоков) агрегирует результаты, полученные на блочном уровне (взаимодействие каждой частицы со всеми) и передает их хосту.

Первоначально для организации нитей (threads) и блоков (blocks) внутри сетки (grid) использовалась *линейная схема* [5] организации процессов, что казалось целесообразным, так как на вход алгоритму подается серия одномерных массивов, и каждая нить осуществляет подсчет взаимодействия между двумя определенными частицами.

В алгоритме используется пошаговая проверка эффективности с возможным возвращением на несколько шагов назад. Авторами проведены компьютерные эксперименты, показавшие высокий процент ожидания процессов, что делает представленную декомпозицию частей неэффективной.

Авторами предложен *блочный* алгоритм, в котором каждый процесс получил задачу по аналитике частицы (взаимодействие выбранной частицы со всеми), а блок процессов – задачу по аналитике группы частиц. Для представленной организации ленточный метод индексирования неприемлем (стал сложным), поэтому для разработки параллельного алгоритма используется блочная схема организации базовых подзадач.

Для оценки корректности этапа разделения вычислений на независимые части используют контрольный список обязательных условий для разбиения: выполненная декомпозиция не увеличивает объем вычислений и необходимый объем памяти; при выбранном способе декомпозиции все процессоры равномерно загружены; выделенных частей процесса вычислений достаточно для эффективной загрузки имеющихся процессоров (с учетом возможности увеличения их количества).

Выделение информационных зависимостей

Учитывая, что передача данных от процессора процессору занимает определенное время и снижает эффективность параллельного алгоритма, ее нужно минимизировать. Оптимальный вариант – перед

каждым шагом моделирования передавать всю информацию о системе на каждый процессор. На рисунке 1 показана схема разделения системы из N объектов на блоки с учетом информационных зависимостей.

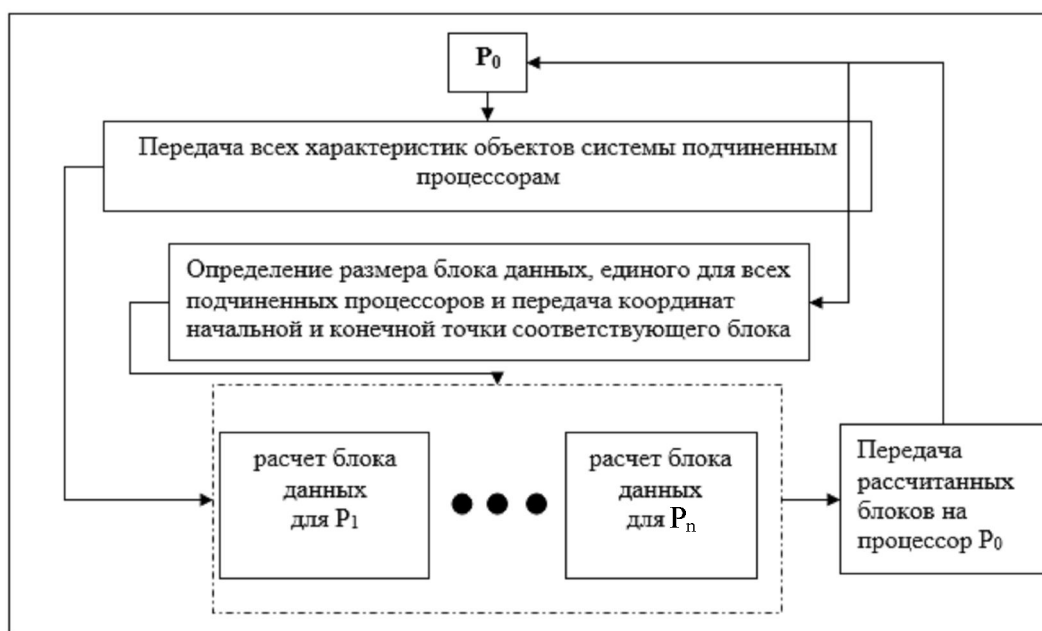


Рис. 1. Разделение системы из N объектов на блоки с учетом информационных зависимостей

Для оценки корректности этапа выделения информационных зависимостей так же, как и для этапа разделения на независимые части, используют контрольный список обязательных условий функционирования: вычислительная сложность подзадач соответствует интенсивности их информационных взаимодействий; интенсивность информационных взаимодействий одинакова для разных подзадач; схема информационного взаимодействия локальна; выявленная информационная зависимость не препятствует параллельному решению подзадач.

Масштабирование и распределение задач по потокам

Параллельный алгоритм называется масштабируемым, если при росте числа процессоров он одновременно обеспечивает увеличение ускорения и сохраняет эф-

фективность использования процессоров. Масштабирование проводится в случае, если количество имеющихся подзадач отличается от числа планируемых к использованию процессоров. Ниже приведены правила агрегации и декомпозиции подзадач, которые параметрически зависят от числа процессоров, применяемых для вычислений.

На рисунке 2 представлен параллельный алгоритм расчета одного шага моделирования системы из N объектов. Алгоритм является оптимальным для системы с количеством взаимодействий N^2 .

Рассмотрим более детально систему с количеством взаимодействий $(N*(N-1))/2$ (каждая частица со всеми без повторов для следующих итераций). На рисунке 3 в матрице отражены системы с количеством взаимодействий N^2 (слева) и $(N*(N-1))/2$ (справа), где объекты обозначаются $O_1...O_n$

и серым цветом закрашены области, где не рассчитываются взаимодействия между объектами.

Представленный алгоритм является оптимальными для однородной информационной среды, но его реализация в гетерогенной среде требует доработки. Во-первых, с числом увеличения частиц увеличивается нагрузка на каждый процессор, время выполнения блока расчетов, связанного с i -ой частицей, увеличивается. Во-вторых, из-за гетерогенности среды более мощные компьютеры простаивают, т.к. выполняют свои расчеты быстрее. Для реше-

ния этих задач создана модель балансировки нагрузки процессоров.

Для оценки правильности масштабирования используют контрольный список обязательных условий функционирования: локальность вычислений не ухудшится после масштабирования имеющегося набора подзадач; подзадачи после масштабирования сохраняют одинаковую вычислительную и коммуникационную сложность; количество задач соответствует числу имеющихся процессоров; правила масштабирования зависят параметрически от количества процессоров.

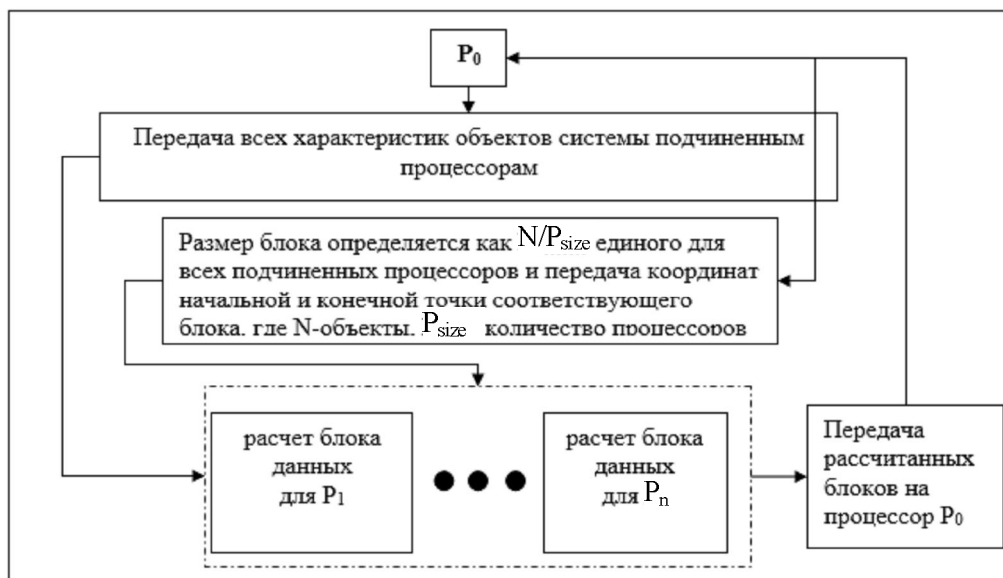


Рис. 2. Алгоритм расчета одного шага моделирования системы из N объектов

	O ₁	O ₂	O ₃	...	O _n
O ₁				...	
O ₂				...	
O ₃				...	
...		
O _n				...	

	O ₁	O ₂	O ₃	...	O _n
O ₁					
O ₂					
O ₃					
...		
O _n				...	

Рис. 3. Количество взаимодействий, выполняемое каждым объектом

Распределение подзадач между процессорами является завершающим этапом разработки параллельного метода. Основ-

ной показатель успешности выполнения данного этапа – эффективность использования процессоров, определяемая как отно-

сительная доля времени, в течение которого процессоры использовались для вычислений, связанных с решением исходной задачи. Пути достижения хороших результатов в этом направлении остаются прежними: обеспечение равномерного распределения вычислительной нагрузки между процессорами и минимизация числа сообщений между ними.

Для оценки правильности распределения использован контрольный список обязательных условий функционирования:

распределение нескольких задач на один процессор не приводит к росту дополнительных вычислительных затрат; существует необходимость динамической балансировки вычислений; процессор-менеджер не является «узким» местом при использовании схемы «менеджер – исполнитель».

Разработанный алгоритм (рис. 4) удовлетворяет всем обязательным пунктам контрольных списков по каждому этапу разработки, также учитываются функциональные и нефункциональные требования.

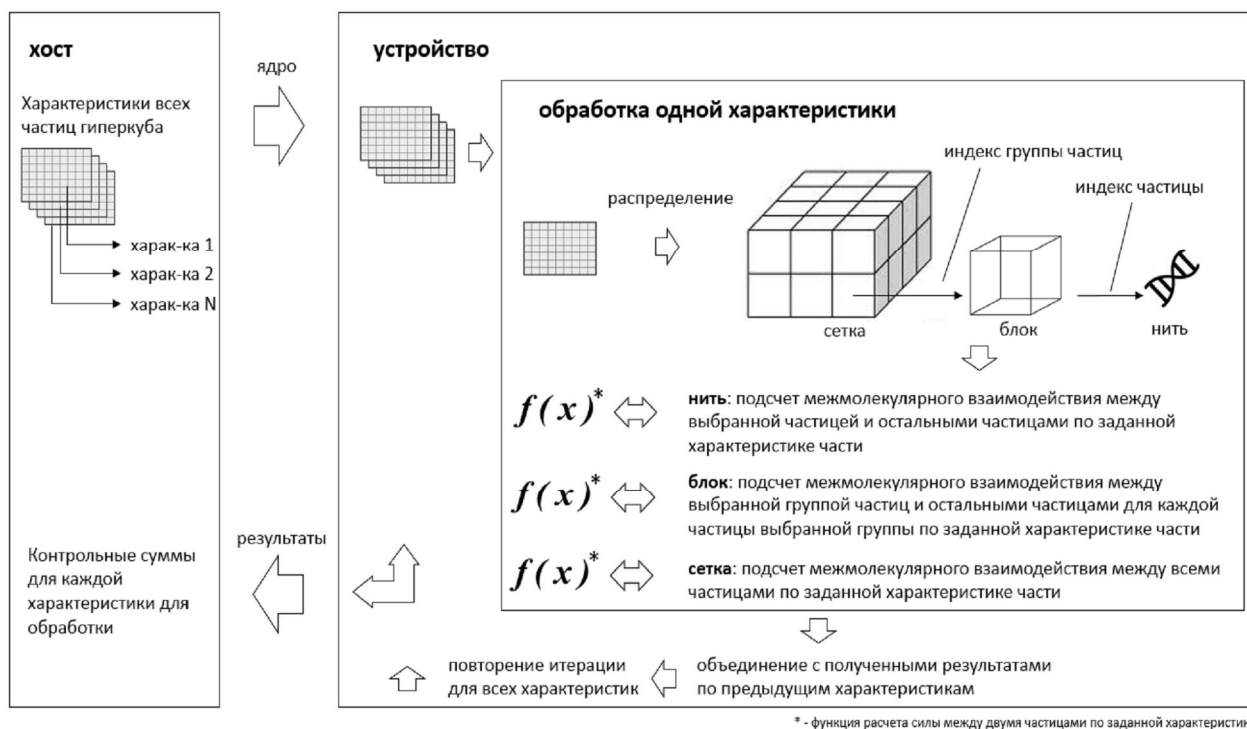


Рис. 4. Алгоритм распределенного вычисления сил межчастичного взаимодействия

Результаты компьютерных экспериментов

Компьютерные эксперименты проводились на центральном процессоре Intel® Core i3-4030U CPU 1.90 GHz с ОЗУ, равной 4,00 Gb, и на графическом процессоре с использованием видеокарты NVidia GeForce 820M для приложения, реализующего

расчет на CUDA. Результаты компьютерных экспериментов, отражающие время вычисления (в секундах) одного программного модуля расчета сил межмолекулярного взаимодействия для параллельного (CUDA) и линейного (host) алгоритмов, представлены в таблице 1.

Таблица 1

		Результаты компьютерных экспериментов						
		Количество частиц						
		1000	5000	10000	50000	100000	500000	1000000
устройство	CUDA	0,2	2	10	345	1287	42234	589711
	host	15	188	1286	53649	218023	7376529	104063567

Таблица 2

Результаты компьютерных экспериментов для разного количества нитей

		Количество частиц						
		1000	5000	10000	50000	100000	500000	1000000
количество нитей	32	0,5	4,7	22	754	1859	60992	3068950
	64	0,2	2,5	10	355	1292	42399	1422241
	128	0,3	2,7	11	370	1294	42453	1017201
	256	0,3	2,3	10	360	1297	42563	784484
	512	0,3	2,3	11	365	1295	42508	652894
	1024	0,2	2,0	10	345	1287	42234	589711

Анализируя полученные результаты, можно сделать предварительное заключение, что реализация параллельного алгоритма дала выигрыш в скорости исполнения модуля подсчета сил межмолекулярного взаимодействия по одной характеристике в более чем 100 раз (усредненное табличное значение ускорения).

В таблице 2 приведены результаты компьютерных экспериментов, отражающих время вычисления одного программного модуля расчета сил межмолекулярного взаимодействия для параллельного

(CUDA) алгоритма при разном количестве потоков (нитей).

Наибольшее ускорение достигается за счет организации блока с максимально возможным количеством нитей. Такая точная организация позволит увеличить размерность моделируемой системы, но с увеличением числа нитей вырастает потребность в дополнительном выделении памяти на графическом процессоре.

На рисунке 5 представлена кривая ускорения разработанного параллельного алгоритма.

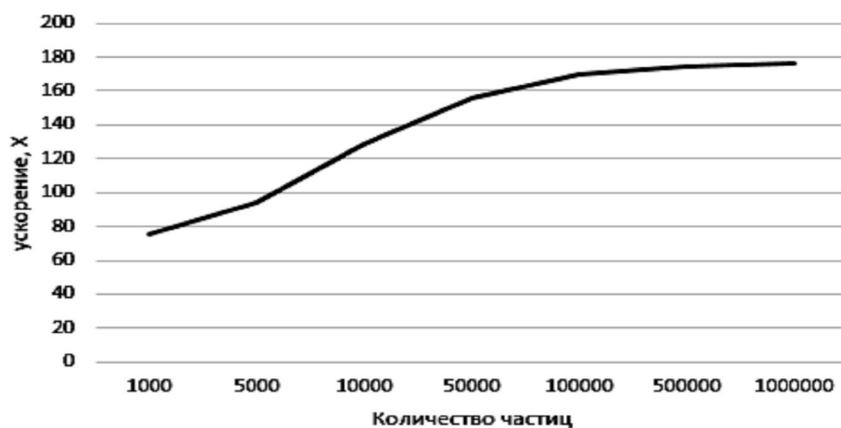


Рис. 5. Кривая ускорения параллельного алгоритма

Пиковое значение скорости наблюдается возле отметки в 50 тыс. частиц, а

минимальное значение выигрыша в производительности составляет 80 раз.

ЛИТЕРАТУРА

1. Воронова Л.И., Григорьева М.А., Воронов В.И., Трунов А.С. Программный комплекс MD-SLAG-MELT информационно-исследовательской системы «Шлаковые Расплавы» версии 10.0. Депонированная рукопись № 29-В2012 26.01.2012.

2. Григорьева М.А., Воронова Л.И. Интеграция XML-данных и вычислительных Fortran-приложений в ИИС «Шлаковые Расплавы» 9.0 // Информационные технологии моделирования и управления. – 2009. – № 1 (53). – С. 106–110.
3. ИИС «MD-SLAG-MELT» [Электронный ресурс] // <http://nano-md-simulation.com/> дата обращения (28.06.2015).
4. Косенко Д.В., Воронова Л.И., Воронов В.И. Разработка программного обеспечения для обработки сложноструктурированных данных научного эксперимента // Вестник Нижневартковского государственного университета. – 2014. – № 3. – С. 45–52.
5. Пилипчак П.Е., Воронова Л.И., Трунов А.С. Модель балансировки нагрузки для параллельного расчета системы N-частиц на гетерогенной кластерной системе // Современные наукоемкие технологии. – 2014. – № 5–2. – С. 218–219.
6. Якововский М. Введение в параллельные алгоритмы: [Электронный ресурс] // www.intuit.ru/ (дата обращения 28.06.2015).
7. Voronova L.I., Grigor'eva M.A., Voronov V.I., Trunov A.S. MD-Slag-Melt Software Package for Simulating the Nanostructure and Properties of Multicomponent Melts // Russian metallurgy (Metally). – 2013. – Vol. 2013. – № 8. – P. 617–627.

REFERENCES

1. Voronova, L.I., Grigorieva, M.A., Voronov, V.I., Trunov, A.S. Programmny complex MD-SLAG-MELT informatsionno-issledovatel'skoy sistemy «Shlakoviye Rasplavy» versii 10.0. Deponirovannaya rukopis № 29-B2012 26.01.2012 [Software Package MD-SLAG-MELT of Slag Melts Research-Information System, Version 10.0. Deposited manuscript № 29 B2012 26.01.2012]. (In Russian).
2. Grigorieva, M.A., Voronova, L.I. Integratsiya XML-dannykh i vychislitelnykh Fortran-prilozheniy v IIS «Shlakoviye Rasplavy» 9.0 [Integration of XML-data and computing Fortran applications in RIS Slag Melts 9.0] // Informatsionniye tekhnologii modelirovaniya i upravleniya. 2009. № 1 (53). P. 106–110. (In Russian).
3. RIS MD-SLAG-MELT [Electronic resource] // <http://nano-md-simulation.com/> (Accessed on June 28, 2015). (In Russian).
4. Kosenko, D.V., Voronova, L.I., Voronov, V.I. Razrabotka programmnogogo obespecheniya dlya obrabotki slozhnokonstruirovannykh dannykh nauchnogo eksperimenta [Development of software for processing complex structured scientific experiment data] // Vestnik Nizhnevartovskogo gosudarstvennogo universiteta. 2014. № 3. P. 45–52. (In Russian).
5. Pilipchak, P.E., Voronova, L.I., Trunov, A.S. Model balansirovki nagruzki dlya parallelnogo rascheta sistemy N-chastits na geterogmniy skasternoy sisteme [Load balancing model for parallel calculation of N-particles on heterogeneous cluster system] // Sovremenniy naukoemkiye tekhnologii. 2014. № 5-2. P. 218–219. (In Russian).
6. Yakobovskiy, M. Vvedeniye v parallelnyye algoritmy [Introduction to parallel algorithms]. [Electronic resource] // www.intuit.ru/ (Accessed on June 28, 2015). (In Russian).
7. Voronova, L.I., Grigor'eva, M.A., Voronov, V.I., Trunov, A.S. MD-Slag-Melt Software Package for Simulating the Nanostructure and Properties of Multicomponent Melts // Russian metallurgy (Metally). 2013. Vol. 2013. № 8. P. 617–627.

*A.S. Trunov, L.I. Voronova, V.I. Voronov
Moscow, Russia*

DEVELOPING PARALLEL ALGORITHM FOR RESEARCH AND INFORMATION SYSTEM OF MD-SLAG-MELT BASED ON CUDA TECHNOLOGY

Abstract. Development of new materials with predetermined properties is a priority trend in modern science. Here computer modeling (CM), including molecular dynamics method allowing us to determine a set of properties (structural, thermodynamic and transport) and study the relationship of nanostructure and physical-chemical properties, are widely applied.

Computer modeling is ensured through the development of automated information systems (AIS) aimed at expanding research boundaries, optimizing and accelerating scientific work. Research and Information System of «Slag Melt» (RIS MD-SLAG-MELT) is a perfect example of such automated systems [1].

RIS's data domain supposes that without using distributed computing it is currently possible to calculate group behavior of the systems containing tens of thousands of particles at most. However, certain tasks of describing nanoscale three-dimensional clusters require increasing the dimensions of the model system to millions of particles.

It is almost impossible to simulate such system on a local computer in a sequential manner due to time input, as the experiment may take several months, hence distributed computing is required.

The authors have developed a parallel algorithm adapting the existing linear algorithm (legacy application) of calculating the forces of intermolecular interaction for distributed computing at CPU and GPU computing device and implemented in RIS MD-SLAG-MELT. The algorithm is based on CUDA distributed parallel programming technology.

Key words: Information and Research Systems (IRS) MD-SLAG-MELT; molecular dynamics; distributed computing; parallel algorithm; CUDA.

About the authors: Artem Sergeevich Trunov¹, Senior Lecturer at the Department of Information Systems and Modeling, Russian State University for Humanities; Lilia Ivanovna Voronova², Head of the Department of Information Systems and Modeling, Russian State University for Humanities, and Professor at the Department of Mathematical Cybernetics and Information Technologies, Moscow Technical University of Communications and Informatics; Vyacheslav Igorevich Voronov³, Associate Professor at the Department of Information Systems and Modeling, Russian State University for Humanities.

Place of employment: Russian State University for Humanities^{1, 2, 3}; Moscow Technical University of Communications and Informatics².

УДК 51-77

О.В. Шульгин
Нижевартовск, Россия

ОБЩАЯ ПОСТАНОВКА ЗАДАЧ АНАЛИЗА И СИНТЕЗА ОРГАНИЗАЦИОННЫХ СИСТЕМ

Аннотация. Предметом настоящего исследования являются организационные системы как системы открытого типа. Работа посвящена проблематике формализации структуры организационных систем в неразрывной связи с их системными функциями (проблематике анализа и синтеза систем).

Целью работы является общая постановка задач анализа и синтеза организационных систем.

Теоретико-методологической базой исследования послужили положения теории систем и методы системного анализа.

В работе рассматривается общая постановка задач анализа и синтеза организационных систем в рамках системного подхода. Каждая задача представлена в качестве последовательно формулируемых подзадач. Ставятся проблемы определения параметров, характеризующих элементарные объекты организационных систем. Дается представление об элементарных функциях, реализуемых данными объектами, а также о взаимосвязях между ними. Последовательно рассматриваются особенности анализа закрытых и открытых систем. Вводится описание взаимосвязей между элементами системы в виде квадратных матриц. Определяется проблематика влияния внешней среды на формирование структуры системы. Дается характеристика зависимых и независимых подсистем. Ставится задача синтеза систем в рамках определения эмерджентной функции (функции системы). Последовательно формулируются подзадачи синтеза через определение математических функций воздействия внешней среды на систему, состояния системы, воздействия системы на внешнюю среду, изменения состояния внешней среды под воздействием системы.

Основным результатом работы является определение ключевых направлений и алгоритма моделирования структурных взаимосвязей организационных систем, имеющих значимость для разработки имитационных моделей функционирования организации и проектирования информационных систем управления организациями.

Ключевые слова: система; системный анализ; моделирование организационных систем.

Сведения об авторе: Шульгин Олег Валерьевич, кандидат экономических наук, доцент кафедры физико-математического образования.

Место работы: Нижевартовский государственный университет.

Контактная информация: 628611, г. Нижевартовск, ул. Дзержинского, д. 11; тел.: 89125392155.
E-mail: shul.oleg.val@mail.ru
