

ИНТЕГРАЛЬНЫЙ ПОДХОД К РАЗРАБОТКЕ АЛГОРИТМИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ЭКЗАФЛОПСНЫХ СУПЕРЭВМ: НЕКОТОРЫЕ РЕЗУЛЬТАТЫ

Б. М. Глинский, И. Г. Черных, И. М. Куликов, А. В. Снытников,
А. Ф. Сапетина, Д. В. Винс

Институт вычислительной математики и математической геофизики СО РАН, 630090, Новосибирск

УДК 004.421, 004.942

Данная работа посвящена применению интегрального подхода к решению вычислительно-сложных задач сейсмике, астрофизики, физики плазмы. Предлагаемый интегральный подход — это методология разработки алгоритмического и программного обеспечения для суперкомпьютеров экзафлопсного уровня, содержащий три связанных этапа: первый этап определяется со-дизайном, под которым мы понимаем адаптацию вычислительного алгоритма и математического метода под архитектуру суперкомпьютера на всех этапах разработки программы; на втором предполагается создание упреждающего алгоритмического и программного обеспечения для наиболее перспективных экзафлопсных суперкомпьютеров на основе имитационного моделирования с целью адаптации алгоритмов под заданную архитектуру суперкомпьютера; третий этап связан с оценкой энергоэффективности алгоритма при различных реализациях на данной архитектуре, либо на различных архитектурах.

Ключевые слова: интегральный подход, со-дизайн, агентное моделирование, энергоэффективность алгоритмов, высокопроизводительные вычисления.

Введение

Современный этап развития суперкомпьютеров характеризуется появлением множества проектов по созданию экзафлопсного суперкомпьютера. До сих пор разработки в данной области проводились различными командами разработчиков в США. Работы в этом направлении проводятся, например, национальными лабораториями Министерства энергетики США: Sandia и Oak Ridge. Существуют аналогичные программы и в Европе: семь европейских стран подписали декларацию Joint Project EuroHPC, направленную на создание сверхсовременного суперкомпьютера. В Японии (RIKEN) сборка и установка такого суперкомпьютера уже началась.

Существует множество международных проектов по разработке системного и прикладного программного обеспечения для экзафлопсных суперкомпьютеров с участием США, стран Европейского Союза, Японии, Китая, России (IESP, G8 EXASCALE, CRESTA и т.д.). В работах [1, 2, 3] приводится обзор различных подходов к разработке научного программного обеспечения для экзафлопсных суперкомпьютеров. В [4] перечислены проблемы, типичные для экзафлопсных систем, а также различные методы решения данных проблем. Следует отметить, что численные алгоритмы разрабатываются медленнее, чем аппаратные средства, поэтому на первом этапе использования экзафлопсных суперкомпьютеров для решения физических задач будут применяться существующие алгоритмы и программы.

В данной работе мы предлагаем использование интегрированного подхода к разработке алгоритмов и программного обеспечения для суперкомпьютеров пета- и экзафлопсного класса [5], который содержит три связанных этапа.

Работа выполнена при частичной финансовой поддержке Российского фонда фундаментальных исследований (код проекта 16-07-00434, код проекта 16-29-15120, код проекта 15-01-00508), гранта Президента РФ (код проекта МК – 1445.2017.9).

Первый этап определяется со-дизайном, под которым мы понимаем адаптацию вычислительного алгоритма и математического метода под архитектуру суперкомпьютера на всех этапах разработки программы. Понятие со-дизайна в контексте математического моделирования физических процессов понимается как построение физико-математической модели явления, численного метода, параллельного алгоритма и его программной реализации, эффективно использующей архитектуру суперкомпьютера. При таком подходе актуальным становится сравнение не только методов решения задачи, но и сравнение физических и математических постановок задачи, с целью создания наиболее эффективной реализации на выбранной вычислительной архитектуре.

На втором этапе предполагается создание упреждающего алгоритмического и программного обеспечения для наиболее перспективных экзафлопсных суперкомпьютеров на основе имитационного моделирования с целью адаптации алгоритмов под заданную архитектуру суперкомпьютера. Основным исследуемым параметром эффективности вычислительного алгоритма при его исполнении на высокопроизводительных суперкомпьютерах — это его масштабируемость. Для сеточных методов численного моделирования критерием масштабируемости является неизменность времени расчета алгоритма при прямо пропорциональном количестве вычислительных узлов увеличению расчетной области. Для моделирования исполнения параллельных программ на большом количестве ядер используется мультиагентный подход [6]. Для исследуемых алгоритмов создаются агенты, имитирующие поведение вычислительных узлов при выполнении соответствующих алгоритмов. Эти агенты, имитируя поведение вычислительных узлов, моделируют вычисления для каждой из задач и отправку данных соседям. Более подробно процесс имитации исполнения программ описан в [7].

Третий этап связан с оценкой энергоэффективности алгоритма при различных реализациях под одну или несколько архитектур. Под энергоэффективностью научных программ в сфере высокопроизводительных вычислений мы понимаем: наиболее эффективное использование каждого ядра процессора или ускорителя вычислений; минимизацию обменов данными между вычислительными узлами; хорошую балансировку программ. Минимизация обменов данными позволяет уменьшить время простоя ядер процессоров или ускорителей. Хорошая балансировка программы позволит равномерно загрузить вычислительную систему. В случае хорошей балансировки программы и стабильной балансировки узлов мы можем сделать набор прогонов программы, который показывает соотношение между потреблением энергии и использованием ядер. Самые энергоэффективные алгоритмы показывают наилучшие значения FLOPS per Watts (Joules / sec). Для тестирования энергоэффективности с помощью инструментария CUDA 7.5 был использован графический процессор Nvidia Tesla K40M.

1 Применение со-дизайна для решения различных физических задач

1.1 Численное моделирование распространения упругих волн в неоднородных трехмерных средах.

Численное моделирование распространения упругих волн в неоднородных трехмерных сложно построенных средах является сложной задачей с точки зрения вычислений, что требует использования эффективных методов распараллеливания и масштабирования алгоритмов. Нередко рельеф реальных геофизических объектов не позволяет установить площадную систему наблюдения. Поэтому для построения трехмерных моделей таких объектов требуется решение обратной задачи путем решения набора прямых задач: при разных значениях упругих параметров гетерогенной среды; при различных геометриях объектов, составляющих модель.

В контексте со-дизайна мы провели сравнение разработанных параллельных реализаций решений задачи динамической теории упругости, записанных в терминах скоростей смещения и напряжения и в терминах смещений, для суперкомпьютеров, оснащенных графическими картами. В качестве области моделирования рассматривается изотропная трехмерная неоднородная упругая сложно построенная среда, представляющая из себя параллелепипед, одна из сторон которого является свободной поверхностью. На этапе выбора численного метода наиболее «гибким» и широко распространенным методом решения задачи динамической теории упругости является метод конечных разностей. Для численного решения уравнений в терминах скоростей смещения и напряжения мы используем известную разностную схему Верье на сдвинутых сетках [9]. Вычисление разностных коэффициентов в этой схеме основано на интегральных законах сохранения. Для решения задачи в терминах смещений мы используем аналогичную схему на сдвинутых сетках [10].

В качестве инструментов распараллеливания программного обеспечения мы выбрали CUDA и MPI. Этап адаптации к архитектуре гибридного кластера, оснащенного графическими процессорами, проводился оди-

наковыми способами для обоих подходов для корректности сравнения [10]. Для распараллеливания мы проводим декомпозицию вычислительной области на слои вдоль одной из осей координат. Каждый слой вычисляется на отдельном узле, где он подразделяется на подслои вдоль другой координатной оси в соответствии с количеством графических ускорителей на узле. Данные передаются между узлами с использованием соответствующих неблокирующих асинхронных функций обменов MPI и асинхронных функций копирования CUDA. Отметим, что данные для обмена имеют одинаковый размер в обоих подходах.

Численные эксперименты показали, что время вычисления смещений и время вычисления скоростей смещения и напряжений при равном числе узлов примерно одинаковы, несмотря на то, что расчет смещений выполняется с большим количеством операций с плавающей запятой на каждом временном шаге. При этом для такого расчета требуется почти в два раза меньше памяти на графических картах. На основе полученных результатов мы отдаем предпочтение подходу, основанному на расчете смещений.

Результаты численного моделирования усеченной модели вулкана Эльбрус представлены на рис. 1. Узнать больше о геофизической модели вулкана и результатах численных экспериментов можно в [11].

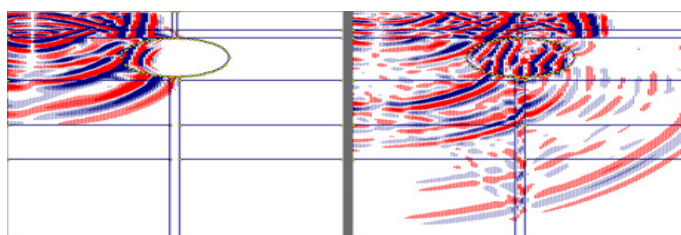


Рис. 1: Результаты численного моделирования усеченной модели вулкана Эльбрус. Снимки волнового поля для компоненты и вектора скоростей смещения представлены в плоскости Oxz в разные моменты времени.

1.2 Многокомпонентная гидродинамическая модель столкновения галактик.

В работе рассматривается многокомпонентная гидродинамическая модель столкновения галактик, учитывающая химиодинамику молекулярного водорода и процессы охлаждения и нагрева. На первом этапе со-дизайна определяется доминантный процесс – гидродинамика, описываемая гиперболическими уравнениями. Численный метод решения уравнений гидродинамики основан на комбинации метода разделения операторов, метода Годунова с модификацией осреднения по Рое и кусочно-параболическом метода на локальном шаблоне вычислений. Переопределенная система позволяет гарантировать неубывание энтропии и производить минимизацию дисбаланса энергий с помощью корректировки вектора скорости. Подробное описание численного метода можно найти в [12].

Код AstroPhi основан на описанных методах. Для его реализации используется архитектура ускорителей Intel Xeon Phi. Для моделирования используется прототип инженерной архитектуры RSC PetaStream с 8 узлами с 64 ускорителем Intel Xeon Phi 7120D. Наши тесты показывают, что 10% общего времени моделирования тратится на операции отправки/получения MPI/OpenMP. Это значение подходит для массивно-параллельных систем.

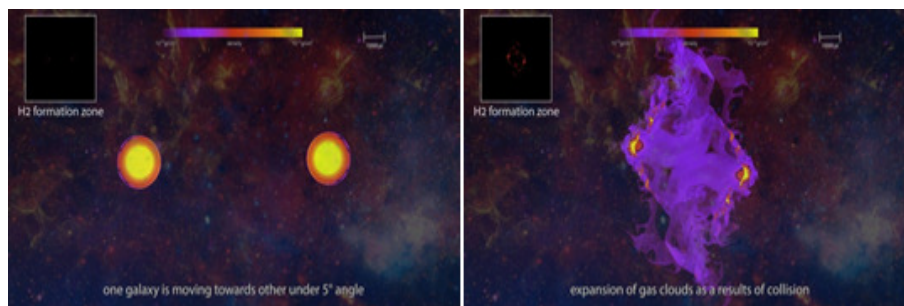


Рис. 2: Химиодинамика столкновения галактик: начальная стадия (слева), разлет газовых облаков после столкновения и зона образования молекулярного водорода (справа).

На рисунке 2 показан разлет двух газовых облаков после столкновения галактик. Одна галактика пролетает через другую с образованием двух газовых облаков и зоны образования молекулярного водорода после удара.

1.3 Взаимодействие мощного электронного пучка с плазмой.

Одной из наиболее интересных проблем в физике высокотемпературной плазмы является резонансное взаимодействие мощного электронного пучка с плазмой.

В настоящей работе используется следующая физическая постановка задачи.

$$\frac{\partial f_{i,e}}{\partial t} + \mathbf{v} \frac{\partial f_{i,e}}{\partial \mathbf{r}} + q_{i,e}(\mathbf{E} + [\mathbf{v}, \mathbf{B}]) \frac{\partial f_{i,e}}{\partial \mathbf{v}} = 0, \quad (1)$$

$$\frac{\partial \mathbf{E}}{\partial t} = \nabla \times \mathbf{B} - \mathbf{j}, \quad \nabla \mathbf{E} = \rho \quad (2)$$

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}, \quad \nabla \mathbf{B} = 0$$

Трехмерное пространство моделирования имеет форму куба. Внутри этой области присутствует модельная плазма. Модельные частицы плазмы распределены равномерно внутри области. Плотность и температура электронов плазмы задается пользователем. Температура ионов считается равной нулю. Электроны пучка также равномерно распределены вдоль области.

Проведено трехмерное кинетическое исследование релаксационных процессов, вызванное распространением электронного пучка в высокотемпературной плазме с использованием системы уравнений Власова-Максвелла.

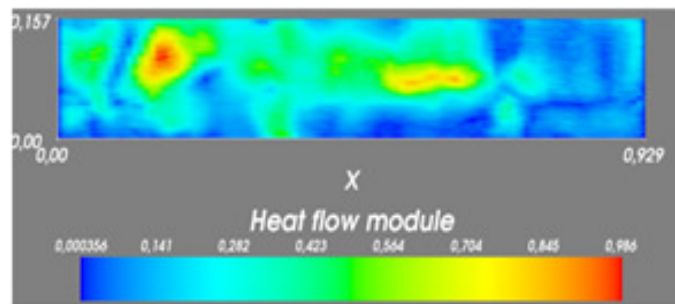


Рис. 3: Моделирование теплового потока в плазме после релаксации пучка.

В данном случае со-дизайн начинается на этапе физического рассмотрения проблемы.

- Отсутствие значительных модуляций плотности позволяет не использовать динамическую балансировку нагрузки.
- Следующим этапом является разработка численного метода. Здесь был выбран метод FDTD, который обеспечивает локальность памяти.
- На этапе выбора архитектуры суперкомпьютера учитываются особенности метода частиц в ячейках.
- На этапе выбора инструментов разработки программного обеспечения со-дизайн заключается в следующем. Для метода частиц в ячейках весьма эффективным является использование технологии CUDA. CUDA дает возможность использовать максимальное количество параллельных процессов и получать максимальную производительность.
- Последний этап со-дизайна — адаптация алгоритма к архитектуре графического процессора. Для этого был осуществлен переход от хранения всех модельных частиц в виде одного большого массива к хранению частиц в виде массивов или списков по ячейкам сетки, что дает значительный прирост производительности.

Эффективность распараллеливания составляет более 90% для 500 узлов. Результаты выполнения полученного кода представлены на рисунке 3

2 Масштабируемость исследуемых алгоритмов

Для исследования масштабируемости параллельных алгоритмов был проведен модельный эксперимент по их исполнению на большом числе вычислительных ядер.

Исходные данные для исследования масштабируемости геофизического кода с помощью имитационной модели получены на кластере НКС-30Т+GPU ЦКП ССКЦ СО РАН для двух подходов к решению задачи. Результаты моделирования представлены на рисунке 4. По результатам моделирования можно сделать несколько выводов: так как схема взаимодействия для обоих подходов (в смещениях и в напряжениях) по сути одна, то их масштабируемость отличается незначительно; исследуемые алгоритмы подходят для исполнения на большом числе вычислительных ядер.

Исходные данные для исследования масштабируемости астрофизического кода с помощью имитационной модели получены на кластере НКС-30Т ЦКП ССКЦ СО РАН и МСЦ, произведено моделирование вычислений данного алгоритма для большого количества ядер (на узлах с Intel Xeon E5-2697 до 5632 узлов, 67 584 ядра; на узлах с GPU Nvidia Kepler до 1024 узлов, 1 048 576 ядер; на узлах с Intel Xeon Phi 7110 до 4096 узлов, 245 760–983 040 ядер) Результаты моделирования представлены на рисунке 5. Из представленного графика можно сделать выводы, что время исполнения алгоритма растет незначительно (около 20%) для 5120 вычислительных узлов и лучшую масштабируемость показали вычислительные узлы с ускорителями Nvidia Kepler K40 и Intel Xeon Phi (native mode).

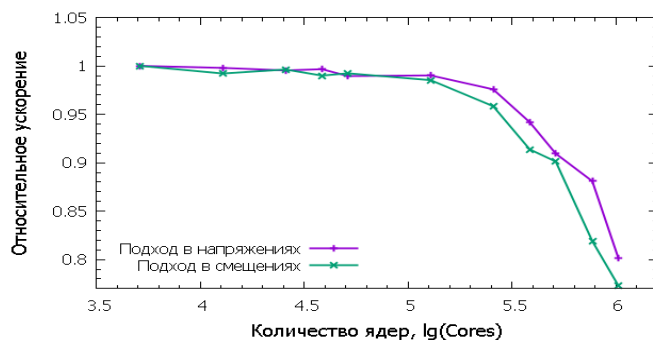


Рис. 4: Исследование масштабируемости геофизического кода для двух подходов к решению (горизонтальная ось – в логарифмическом масштабе).

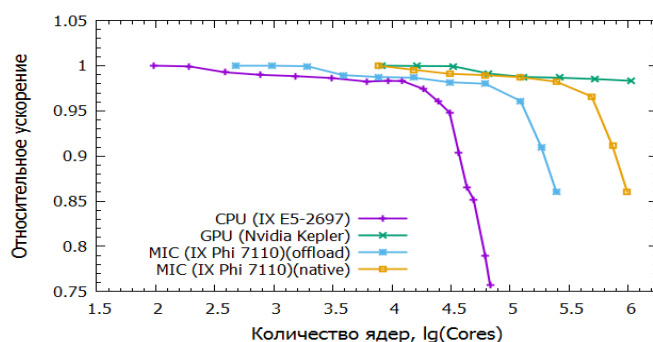


Рис. 5: Исследование масштабируемости астрофизического кода на различных архитектурах (горизонтальная ось – в логарифмическом масштабе).

Исходные данные для исследования масштабируемости кода физики плазмы с помощью имитационной модели получены на кластере НКС-30Т ЦКП ССКЦ СО РАН и МСЦ МФЦ, произведено моделирование вычислений данного алгоритма для большого количества ядер М (на узлах с Intel Xeon E5-2697 до 3072 узлов, 36 864 ядра; на узлах с GPU Nvidia Tesla 2090М до 1536 узлов, 786 432 ядер; на узлах с Nvidia Kepler K40 до 1536 узлов, 4.4 миллиона ядер). На Рис. 6 представлены результаты имитационного моделирования кода физики плазмы. Выводы: существенное увеличение коммуникационных взаимодействий увеличивает

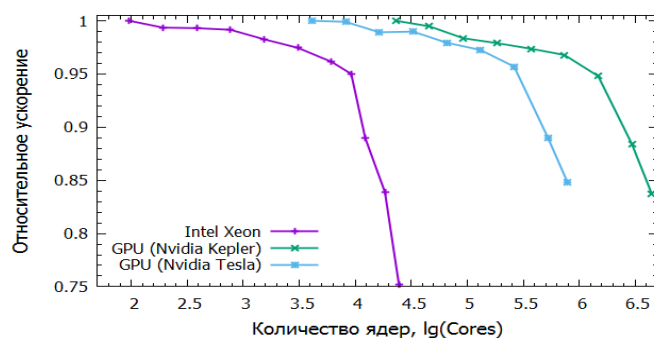


Рис. 6: Исследование масштабируемости кода физики плазмы на различных архитектурах (горизонтальная ось — в логарифмическом масштабе).

на 58% время выполнения кода для 3072 вычислительных узлов с Intel Xeon CPUs и код физики плазмы хорошо масштабируется на вычислительных узлах с ускорителями Nvidia Kepler K40 GPUs.

3 Исследование энергоэффективности алгоритмов

Для нашего астрофизического кода нам удалось уменьшить время MPI операций до 7–8 % от общего времени выполнения программы и добиться уровня разбалансировки процессов не более 2–3% между всеми нитями процессов. Такие показатели позволили получить 75% эффективность (weak scalability) распараллеливания на 224 Intel Xeon Phi (более 50K ядер).

Применяя этот же подход для кода физики плазмы была получена 92 % эффективность распараллеливания на 500 Tesla GPUs (более 250K ядер).

Принимая во внимание особенности геофизического кода, мы достигли энергоэффективности 9 GFLOPS/Вт и 12 GFLOPS/Вт для решения задачи в напряжениях и в смещениях соответственно на Nvidia Tesla K40M GPU. Мы также достигли энергоэффективности 4,3 GFLOPS/Вт и 4,5 GFLOPS/Вт для тех же подходов на Nvidia Tesla 2090M GPU без изменения исходного кода.

Заключение

В этой статье предлагается комплексный подход к разработке алгоритмов и программного обеспечения для решения физических задач, требующих большого объема вычислений. В контексте со-дизайна мы провели сравнение разработанных параллельных реализаций решения задачи динамической теории упругости, записанной в разных постановках для гибридных кластеров, оснащенных графическими картами. Метод Годунова с модификацией усреднения по Рою и кусочно-параболический метод на локальном трафарете был выбран из нескольких разных подходов к решению астрофизической задачи. Аналогичный подход был применен и для проблемы физики плазмы. Масштабируемость полученных алгоритмов была проверена с использованием системы моделирования AGNES[8]. В нашем случае в процессе моделирования можно определить оптимальное количество ядер для конкретной архитектуры. Это позволяет исследовать масштабируемость алгоритма, не прибегая к прямым трудоемким вычислениям. Энергоэффективность алгоритма для геофизической задачи была рассмотрена на суперкомпьютерах, оснащенных графическими процессорами Tesla 2090M и K40M.

В результате был разработан набор параллельных программ для решения физических задач на основе описанного подхода. Он способен выполнять 3D-моделирование в приемлемое время, при условии достаточности ресурсов

Список литературы

- [1] Reed D.A., Dongarra J.: Exascale computing and big data. Comm. of the ACM 58 (7), 56–68 (2015).
- [2] Keyes D.E.: Exaflop/s: The why and the how. C.R. Mechanique 339, 70–77 (2011).

- [3] Asanovic K., Bodik R., Demmel J., Keaveny T., Keutzer K., Kubiawicz J., Morgan N., Patterson D., Sen K., Wawrzynek J., Wessel D., Yelick K.: A view of the parallel computing landscape. Comm. ACM. 52, 56–67 (2009).
- [4] Sterling T.: Achieving scalability in the presence of asynchrony for exascale computing. Adv. In Parall. Comp 24, 104–117 (2013).
- [5] Б.М. Глинский, И.В. Куликов, А.В. Снытников, И.Г. Черных, Д.В. Винс Многоуровневый подход к разработке алгоритмического и программного обеспечения экзафлопсных суперЭВМ // Вычислительные методы и программирование: новые вычислительные технологии. – 2015. Т.16, №4, – С. 543–556.
- [6] Wooldridge M.: Introduction to MultiAgent Systems. JOHN WILEY & SONS, LTD, England (2002).
- [7] Kulikov, I., Chernykh, I., Glinsky, B., Weins, D., Shmelev, A. Astrophysics simulation on RSC massively parallel architecture //Proc. 2015 IEEE/ACM 15th Int. Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2015. IEEE Press, 2015.1131–1134.
- [8] Podkorytov D., Rodionov A., Choo H.: Agent-based Simulation System AGNES for Networks Modeling: Review and Researching. In: Proc. of the 6th Int. Conference on Ubiquitous Information Management and Communication (ACM ICUIMC 2012), ISBN 978-1-4503-1172-4, pp. 115. ACM (2012).
- [9] Bihn M., Weiland T.: A Stable Discretization Scheme for the Simulation of Elastic Waves. In: Proceedings of the 15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics (IMACS 1997). vol. 2, pp. 75–80. Berlin (1997) .
- [10] Sapetina A.F.: Supercomputer-aided comparison of the efficiency of using different mathematical statements of the 3D geophysical problem. The Bulletin of NCC. Series: Numerical Analysis 18, 1–9 (2016).
- [11] Glinskii B.M., Martynov V.N., Sapetina A.F.: 3D Modeling of Seismic Wave Fields in a Medium Specific to Volcanic Structures. Yakutian Mathematical Journal. 22(3), 84–98 (2015).
- [12] Kulikov I., E. Vorobyov E.: Journal of Computational Physics. 317, 318–346 (2016).

*Глинский Борис Михайлович — д.т.н., зав. лаборатории Института
вычислительной математики и математической геофизики СО РАН;
e-mail: gbm@sscc.ru;*

*Черных Игорь Геннадьевич — к.ф.-м.н., ст.науч.сотр. Института
вычислительной математики и математической геофизики СО РАН;
e-mail: chernykh@parbz.sssc.ru;*

*Куликов Игорь Михайлович — к.ф.-м.н., науч.сотр. Института
вычислительной математики и математической геофизики СО РАН;
e-mail: kulikov@ssd.sssc.ru;*

*Снытников Алексей Владимирович — к.ф.-м.н., науч.сотр. Института
вычислительной математики и математической геофизики СО РАН;
e-mail: snytav@ssd.sssc.ru;*

*Сапетина Анна Федоровна — инженер Института
вычислительной математики и математической геофизики СО РАН;
e-mail: afsapetina@gmail.com;*

*Винс Дмитрий Владимирович — к.т.н., мл.науч.сотр. Института
вычислительной математики и математической геофизики СО РАН;
e-mail: vins@sscc.ru.*

Дата поступления — 31 мая 2017 г.