УДК 519.683 doi 10.26089/NumMet.v19r432

ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ В ГИДРОГЕОЛОГИЧЕСКОМ РАСЧЕТНОМ КОДЕ GeRa: ОРГАНИЗАЦИЯ И ЭФФЕКТИВНОСТЬ

И. В. Капырин[1], И. Н. Коньшин[2], Г. В. Копытов[3], В. К. Крамаренко[4]

Рассмотрена структура, функциональность и организация параллельных вычислений расчетного кода GeRa, применяемого для моделирования гидрогеологических задач. Описаны принципы запуска на счет многопроцессорных заданий на удаленном кластере из графического интерфейса GeRa. Приведены результаты параллельных расчетов для трех репрезентативных моделей и проведен анализ их параллельной эффективности.

Ключевые слова: геофильтрация, геомиграция, расчетный код, параллельные вычисления, ускорение.

1. Введение. Расчетный код GeRa [1] разрабатывается в Институте проблем безопасного развития атомной энергетики (ИБРАЭ РАН) и Институте вычислительной математики (ИВМ РАН) для решения задач, связанных с безопасным захоронением радиоактивных отходов (РАО), оценки безопасности объектов ядерного наследия и защиты подземных вод от химического загрязнения. Программный комплекс GeRa является интегральным кодом, т.е. позволяет проводить полный цикл работ, начиная от создания геологической модели на основе данных полевых и лабораторных исследований и заканчивая расчетом доз для населения по основным путям облучения. Первая версия кода, названная GeRa/V1, была аттестована Ростехнадзором в 2018 г. В настоящее время разрабатывается вторая версия кода, в которой уже реализованы модели следующих процессов:

— насыщенная (в напорных и безнапорных условиях) и ненасыщенная фильтрация;

— адвективно-диффузионно-дисперсивный перенос растворенных веществ в однородных и двупористых средах;

— радиоактивный распад компонентов с учетом цепочек распада;

— равновесная сорбция компонентов, определяемая изотермой Генри с, возможно, переменным коэффициентом распределения Kd;

— течения растворов переменной плотности и вязкости;

— перенос с химическими взаимодействиями;

— тепловыделение при радиоактивном распаде и тепловая конвекция.

Решение прикладных гидрогеологических задач требует учета большого количества химических и физических процессов, больших временны́х (до миллиона лет) и пространственных масштабов моделей, высокой детализации сеток для обеспечения точности расчетов. Разработка расчетных кодов для массивно-параллельного гидрогеологического моделирования является актуальной задачей. Среди них следует отметить разрабатываемые в США коды Amanzi [2], PFLOTRAN [3] и разрабатываемый преимущественно в Европе код OpenGeoSys [4]. Учитывая современные тенденции, расчетный код GeRa ориентирован на использование неструктурированных многогранных расчетных сеток [5], методов конечных объемов для дискретизации задач и распараллеливание вычислений на основе MPI.

[1] Институт проблем безопасного развития атомной энергетики РАН, Большая Тульская ул., д. 52, 115191, Москва; зав. лабораторией, e-mail: ivan.kapyrin@gmail.com

[2] Вычислительный центр им. А. А. Дородницына РАН, ул. Вавилова, 40, 119333, Москва; науч. сотр., e-mail: igor.konshin@gmail.com

[3] Балтийский федеральный университет им. И. Канта, ул. А. Невского, 14, 236016, Калининград; доцент, e-mail: gkopytov@yandex.ru

[4] Институт вычислительной математики им. Г. И. Марчука РАН, ул. Губкина, 8, 119333, Москва; мл. науч. сотр, e-mail: kramarenko.vasiliy@gmail.com

© Научно-исследовательский вычислительный центр МГУ им. М. В. Ломоносова

В настоящей статье описана методика создания параллелизуемого расчетного кода на основе общедоступных библиотек и программного обеспечения, использованная при разработке GeRa. Представлены технические аспекты организации взаимодействия пользовательского графического интерфейса с удаленным вычислительным кластером. На ряде репрезентативных задач (модели полигона закачки жидких радиоактивных отходов (ЖРО) и переноса в геологическом пласте с учетом химических взаимодействий) анализируется эффективность параллельных расчетов.

2. Структурные и функциональные особенности расчетного кода GeRa. В этом разделе с различных сторон рассмотрены структурные, функциональные и другие особенности кода GeRa, чтобы, с одной стороны, показать организацию вычислений конкретного расчетного кода, а с другой стороны, выделить общие черты, характерные для разработки любого программного обеспечения для решения задач математической физики. Это может быть полезно как пользователям кода GeRa, так и разработчикам своих собственных расчетных кодов.

Перечислим общие *функциональные* возможности, которые должен обеспечивать расчетный код с точки зрения пользователя:

— связать все расчетные модули воедино с помощью графического интерфейса (GUI, Graphical User Interface);

— вводить данные пользователя (геометрические, геологические, физико-химические, а также данные, специфические для конкретной области моделирования);

— включать и исключать из рассмотрения отдельные моделируемые физические процессы;

— обеспечивать и переключать различные режимы дискретизации и расчета;

— визуализировать результаты расчета.

Специфические особенности при моделировании гидрогеологических задач в расчетном коде GeRa, используемом для обоснования безопасности захоронения РАО [1]:

— вода является основным объектом моделирования: учет поверхностных водоемов (реки, озера, дрены и др.), учет подземной фильтрации и переноса различных компонентов;

— изменение во времени граничных условий и источников (работа нагнетательных и разгрузочных скважин по расписанию, расписания для учета внешнего поступления воды через осадки на поверхность Земли, через реки и др.);

— учет химических реакций (использование общедоступного пакета iPHREEQC [6, 7]);

— учет радиоактивного распада компонентов и возможных цепочек распада;

— возможность мониторинга процесса моделирования для верификации расчетов;

— возможность расчета дозовых нагрузок на население.

С точки зрения разработчика расчетный код представляет собой совокупность различных программных модулей, схемы передачи данных между ними, а также совокупности программных средств, используемых для создания расчетного кода. Рассмотрим расчетный код GeRa с этой точки зрения.

Перечислим основные программные модули, входящие в GeRa, которые являются *общими* для большинства расчетных кодов моделирования задач математической физики:

— графический интерфейс, который, с одной стороны, предоставляет возможность пользователю задать все необходимые для расчета данные, а с другой стороны, связывает воедино все интерфейсные, вспомогательные и расчетные модули;

— генератор расчетных сеток, который отражает не только геометрию рассматриваемой области, но и другие специфики различных приложений и особенностей решения конкретных задач. Большинство расчетных кодов используют свои собственные генераторы сеток. Например, в расчетном коде GeRa используются два генератора расчетных сеток: треугольно призматический, наиболее адекватно отражающий особенности геологических слоев, а также гексаэдральный со сколотыми ячейками, используемый при необходимости локального сгущения сеток в выбранных областях, в том числе и по вертикали;

— модуль UserData обработки данных пользователя, который переносит общие данные модели на расчетную сетку и служит для выполнения запросов от модуля дискретизации относительно характеристик конкретных ячеек расчетной сетки. Этот модуль использует всю специфику решаемой задачи и пользовательских данных, делая модуль дискретизации более универсальным и ограждая его от привязки к конкретным данным пользователя и специфики решаемых задач;

— модуль дискретизации, который выполняет дискретизацию уравнений, описывающих физические процессы, вовлеченные в моделирование. Например, в расчетном коде GeRa используются несколько видов дискретизации: простейшая двухточечная схема дискретизации, нелинейный метод конечных объемов с монотонной аппроксимацией, многоточечная O-схема аппроксимации, явные и неявные аппроксимации уравнений, схемы первого и второго порядков для адвекции. Пользователь может самостоятельно выбрать одну из реализованных схем дискретизации;

— модуль решения линейных систем, которые получаются после выполнения дискретизации. Обычно в процессе моделирования происходит многократное решение линейных систем, что бывает обусловлено, например, выполнением явных или неявных шагов по времени. Обычно работа этого модуля бывает наиболее ресурсозатратной, поэтому в расчетном коде GeRa используется широкий набор линейных решателей: как линейные решатели из программной платформы INMOST (Integrated Numerical Modelling and Object-oriented Supercomputing Technologies) [8, 9], так и решатели из общедоступного пакета PETSc (Portable, Extensible Toolkit for Scientific computation) [10];

— модуль визуализации полученного решения, который может производиться как через внешнюю программу ParaView [11], так и с помощью средств, полностью реализованных в рамках расчетного кода GeRa.

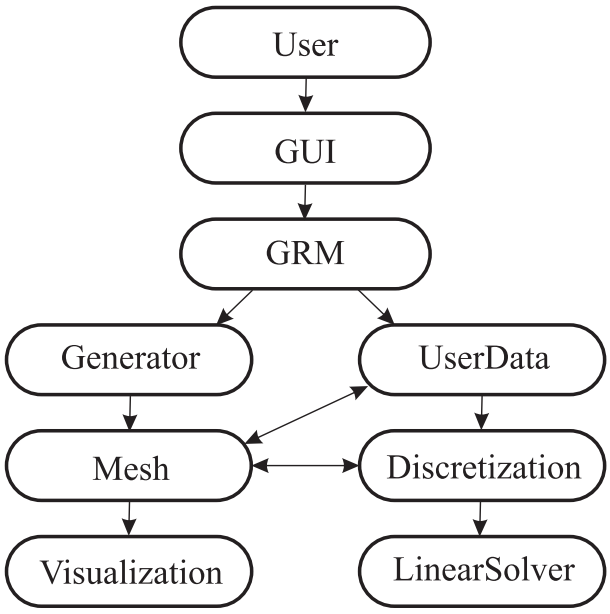


Рис. 1. Взаимодействие пользователя
и модулей GeRa

Одной из важнейших характеристик работы расчетного кода является *взаимодействие* программных модулей и передача данных от одного модуля к другому. В расчетном коде GeRa это взаимодействие схематически можно описать следующим образом (рис. 1):

— пользователь (User) с помощью графического интерфейса (GUI) задает расчетную модель (GRM);

— с помощью геометрических данных этой модели модуль генератора (Generator) расчетных сеток производит построение расчетной сетки (Mesh);

— модуль UserData обработки данных пользователя производит пересчет данных пользователя для каждой ячейки сетки;

— модуль дискретизации (Discretization) обращается за данными и к расчетной сетке, на которой, например, хранится само решение, и к модулю UserData, для получения различных характеристик, например свойств материалов данной ячейки или внешних воздействий на нее;

— полученная в результате дискретизации линейная система решается в модуле LinearSolver, возвращая решение в модуль дискретизации для записи его в расчетную сетку;

— после завершения процесса моделирования решение из расчетной сетки поступает в модуль визуализации (Visualization) для его анализа.

И, наконец, перечислим *программные средства*, используемые разработчиками расчетного кода GeRa:

— для разработки программного обеспечения используется язык C++ и кроссплатформенный фреймворк Qt [12];

— для работы с распределенными сеточными данными, а также для решения линейных систем используется программная платформа INMOST вместе с общедоступными пакетами ParMetis [13] и PETSc;

— для расчета химических взаимодействий используется модуль iPHREEQC;

— для визуализации данных пользователя и расчетных данных используется открытая библиотека Visualization ToolKit (VTK) [14];

— для удобства и безопасности работы разработчиков расчетного кода GeRa используется система контроля версий SVN [15].

Таким образом, почти все программные средства, используемые для разработки кода GeRa, включая SVN, VTK, iPHREEQC, INMOST, ParMetis и PETSc, являются *открытыми* (при соблюдении условий лицензии). Это подтверждает тот факт, что, используя только открытое программное обеспечение, можно разрабатывать мощные и эффективные приложения для решения задач математической физики.

Необходимо отметить, что другая важная особенность расчетного кода GeRa, которая была заложена еще при его разработке, — это использование параллельных вычислений для повышения скорости и точности расчетов [16, 17]. Основой для организации параллельных вычислений в коде GeRa является открытая программная платформа INMOST, разрабатываемая в ИВМ РАН. Анализ результатов эффективности ее применения в численных экспериментах приводится в разделе 5. Вопрос организации запуска параллельного расчета на удаленном кластере рассматривается в нижеследующем разделе 3.

3. Запуск задания на удаленном кластере в графическом интерфейсе GeRa. Для запуска параллельных заданий на кластере графический интерфейс GeRa взаимодействует с компонентом системы, называемом MPI-сервером. Взаимодействие происходит по TCP/IP (Transmission Control Protocol/Internet Protocol) соединению по специально разработанному протоколу. Компонент MPI-сервер может принимать и обслуживать подключения от многих клиентов GeRa одновременно. Расчетный код GeRa может подключиться к MPI-серверу тремя способами.

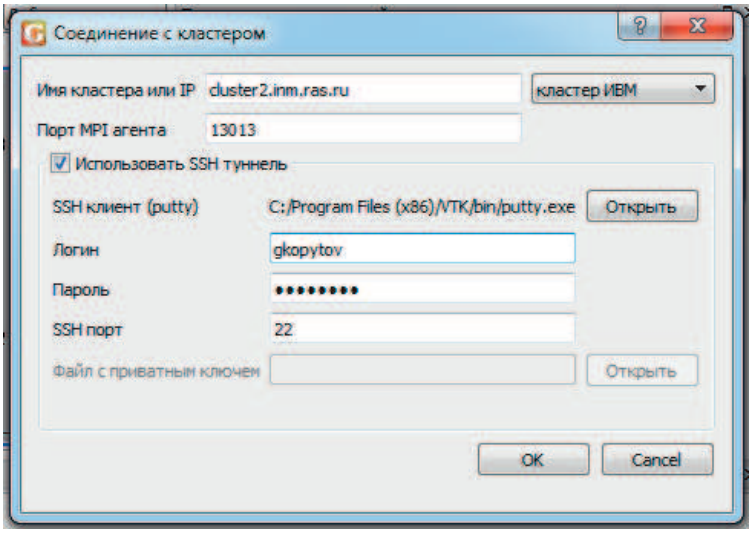
Первый — прямое подключение на порт 13013 из локальной сети кластера, если такое подключение не блокируется межсетевым экраном.

Второй способ — подключение через SSH (Secure SHell) туннель с использованием логина и пароля. Этот способ используется для взаимодействия интерфейса GeRa и удаленного кластера, когда клиент не находится в локальной сети и порт 13013 закрыт на доступ межсетевым экраном.

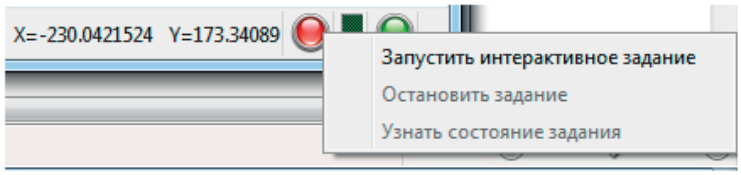
Третий способ подключения — через SSH-туннель с использованием приватного ключа.

В нашей работе будет рассматриваться второй способ подключения, который используется при работе с кластером ИВМ РАН. На рис. 2 показан порядок действий при подключении интерфейса GeRa к кластеру ИВМ РАН.

Сначала необходимо установить VPN-соединение (Virtual Private Network) с локальной сетью ИВМ РАН. Это необходимо, потому что SSH-доступ сконфигурирован таким образом, что обращение возможно только из локальной сети. После этого необходимо создать или загрузить модель, сгенерировать расчетную сетку и вызвать диалог подключения к кластеру через меню “Численная модель/Подключиться к кластеру”, как показано на рис. 2а. Диалог подключения к кластеру позволяет выбрать либо одну из предопределенных конфигураций кластера, либо ввести IP (Internet Protocol) или DNS (Domain Name System) имя кластера и номер порта вручную. Если при подключении использована опция “Использовать SSH-туннель”, то диалог проверяет, что программа “putty.exe”, которая является SSH-клиентом на



a)



b)

Рис. 2. Диалог подключения к кластеру (a); управляющие элементы для запуска заданий (b)

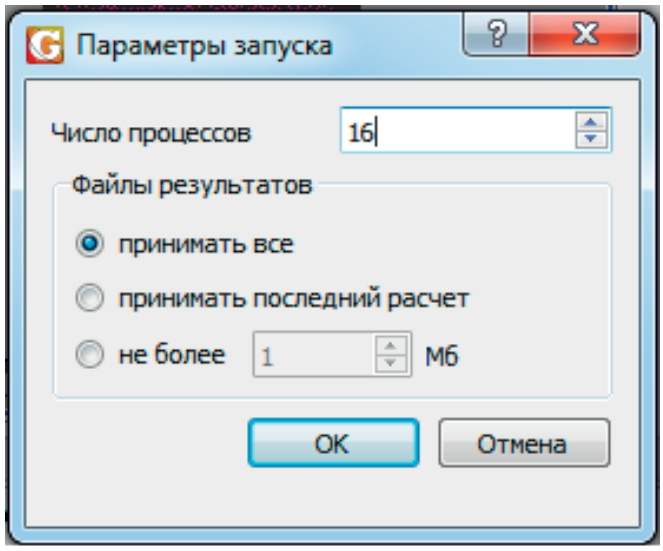
Windows, доступна в переменной “PATH”. Если программа putty не доступна, то пользователь должен указать полный путь до нее с помощью кнопки “Открыть”, иначе подключение с использованием SSH-туннеля будет невозможно. Пользователь также должен указать логин и пароль, который был выдан ему администратором для доступа на кластер.

Получив необходимую информацию, диалог подключения запускает программу putty и организует туннель между портом 13013 локальной машины и кластером, который прозрачно переадресует все подключения и трафик между локальным портом 13013 и удаленным портом на кластере. Помимо переадресации трафика, туннель также выполняет функцию начального запуска MPI-сервера, тем самым устраняя необходимость использования inetd или аналогичной службы для запуска MPI-сервера при первом обращении. MPI-сервер работает по принципу демона UNIX — он запускается в единственном экземпляре и в дальнейшем обслуживает все входящие подключения. Поскольку запуск MPI-сервера на кластере при первом подключении может занять некоторое время, первые попытки подключиться на локальный порт 13013 могут оказаться неуспешными. В этой связи, предусмотрен механизм автоматического повторного подключения при неудаче каждые 5 секунд до 4 попыток, и соединение считается установленным, если клиент GeRa и MPI-сервер обменялись сообщениями “HELLO”.

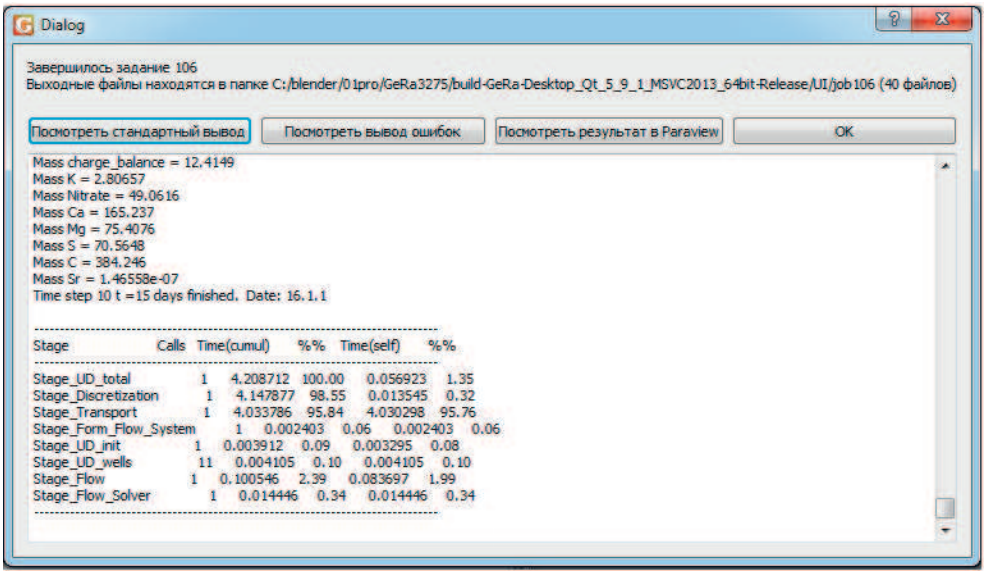
При установлении успешного соединения в правом нижнем углу интерфейса GeRa возникает управляющий элемент в виде цветового индикатора “светофора”, как показано на рис. 2б. Зеленый свет индикатора означает, что соединение установлено нормально, и пользователь может запускать задания на кластере. Красный свет светофора означает, что соединение не установлено по какой-то причине, и эту причину можно узнать, наведя курсор на светофор. Через некоторое время под курсором появляется текст-подсказка с объяснением причины. Управляющие элементы (“светофоры”) создаются при подключении к кластеру, причем их может быть несколько, если пользователь подключился к различным кластерам. Подключение через тоннель при этом возможно только к одному из кластеров. Один из “светофоров” отвечает за запуск параллельных заданий непосредственно на локальной машине. Управление запуском заданий на локальной машине и на кластере унифицировано. По нажатию правой кнопки мыши на соответствующем светофоре возникает опция запустить задание на указанном числе процессоров, а при наличии запущенного задания можно узнать его состояние или завершить задание (рис. 2б).

При запуске задания на кластере пользователь должен выбрать несколько опций, как показано на рис. 3а. Во-первых, он должен указать число процессоров, на которых будет распараллелено решение

задачи. Кроме того, он может ограничить число передаваемых файлов результатов после расчета. По умолчанию файлы результатов передаются все без ограничения, но пользователь может указать, что ему интересен лишь последний расчет, либо он может ограничить по объему количество передаваемых файлов. В последнем случае сервер передаст клиенту группы расчетных файлов, начиная с последней группы, но с суммарным объемом, ограниченным указанным количеством.



a)



b)

Рис. 3. Параметры запуска задания (a); диалог просмотра результатов (b)

Запуск задания на кластере включает в себя передачу необходимых файлов и формирование пакета PBS (в случае кластера ИВМ РАН) для запуска задания на счет. В число передаваемых файлов входит сама модель, а также, например, файлы с начальными условиями. Если модель использует в расчете химические взаимодействия, то на кластер также передается файл с базой данных химических взаимодействий.

По завершению задания у пользователя появляется диалог, показанный на рис. 3б. В этом диалоге пользователь может посмотреть файлы стандартного вывода и файл ошибок задания, а также запустить просмотр результатов во внешней программе ParaView [11].

4. Описание численных моделей. Для демонстрации работы GeRa в параллельном режиме было выбрано три модели. В первых двух из них моделируются гидрогеологические процессы на реальном объекте, полигоне закачки жидких радиоактивных отходов (ЖРО) “Северный” [18]. На полигоне “Север-

ный” начиная с 1967 г. проводится закачка ЖРО в глубоко залегающие пласты-коллекторы с высокими фильтрационными свойствами. Структурно объект содержит 10 геологических слоев, имеющих выклинивания. Горизонтальный размер модели составляет около 27 км, вертикальный — приблизительно 600 м. Водный баланс определяется инфильтрационным питанием и разгрузкой в реки, расположенные на верхней и боковой границах модели.

В первой модели (далее называемой “Pflow”) решается стационарная задача фильтрации на полигоне в безнапорном режиме. Ее решением является распределение напора и уровня подземных вод в расчетной области. Стационарная задача при этом решается методом установления, для этого требуется 24 итерации. На каждой итерации метода установления решается нелинейная задача с помощью метода Ньютона (в среднем требуется 4 итерации). Отметим, что указанное число итераций методов Ньютона и установления не зависят от количества используемых процессоров, в отличие от числа итераций, требуемого для решения возникающих линейных систем. На рис. 4 показано рассчитанное распределение поля напора грунтовых вод на площадке полигона. Четко прослеживается дренирующая роль рек на поверхности (река Большая Тель с ее притоками) и на северной границе модели (река Кан). Им соответствуют низкие значения напора, отмеченные голубым цветом.

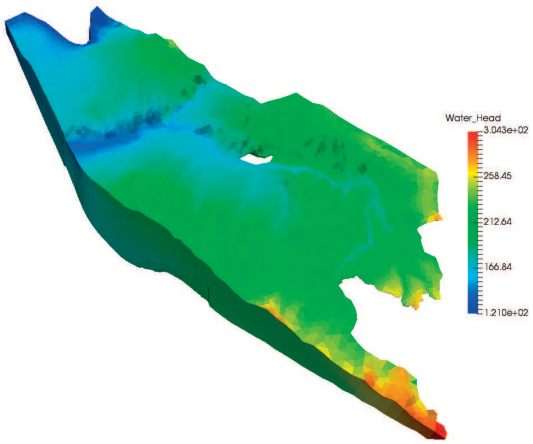


Рис. 4. Распределение напора грунтовых вод в расчетной области для модели Pflow, масштабирование по вертикали в 10 раз

Результаты этого расчета служат начальными условиями для нестационарной задачи фильтрации и миграции радионуклидов на полигоне, которая решается в рамках второй модели, далее обозначаемой как “Ptran”. Учитывается множество гидрогеологических объектов: нагнетательные и разгрузочные скважины, поверхностная гидросеть. Нестационарный характер фильтрационного процесса определяется динамикой работы нагнетательных и разгрузочных скважин. Моделируются процессы фильтрации в безнапорном режиме, адвективно-диффузионно-дисперсионного переноса в среде с двойной пористостью, сорбции и радиоактивного распада. Расчет выполняется для двух компонентов ЖРО — химически нейтрального нитрат-иона и сорбируемого стронция-90. Используются полностью неявные методы конечных объемов для дискретизации задачи. Сетка для обеих моделей полигона содержит 480 350 ячеек, она строится с помощью встроенного в GeRa треугольно-призматического генератора. При ее построении использована адаптация к скважинам: шаг сетки возле каждой скважины по горизонтали составляет 10 метров, в то время как на удалении от источника загрязнения (нагнетательных скважин) шаг сетки достигает приблизительно 600 метров. Для дискретизации по вертикали используется 17 сеточных слоев. Расчет выполняется на временной период с 1967 по 2010 гг. с временны́м шагом 10 дней. Результаты расчета динамики ареалов загрязнения показаны на рис. 5.

Рис. 5. Расчетные ареалы распространения загрязнения для модели Ptran в I (синий) и II (зеленый) в пластах-коллекторах на 2010 г., масштабирование по вертикали в 10 раз

Третья модель “Pchem” — двумерная задача о переносе с детальным расчетом химических взаимодействий, представленная в работе [19]. В ограниченном квадратном пласте через скважины, расположенные в углах этого квадрата, производится закачка растворов с различным содержанием нитрата натрия, но одинаковым содержанием стронция, с дебитом каждой скважины 10 м^3/сут. В центре квадрата расположена разгрузочная скважина, расход которой (40 м^3/сут.) сбалансирован с четырьмя вышеупомянутыми нагнетательными. В закачиваемых растворах также содержатся натрий, кальций, хлор и ряд других элементов и соединений. Порода пласта содержит ионообменные фазы, интерес в данном эксперименте представляет влияние концентрации нитрата натрия в закачиваемом растворе на сорбцию радиоактивно-

го стронция на породе. Для расчета химических взаимодействий в системе раствор–порода используется модуль iPHREEQC [6, 7] (специализированная версия программы PHREEQC, интерфейс для интеграции с геомиграционными кодами), интегрированный в GeRa. Проводится моделирование на 300 суток. В одном сеточном слое по оси Z треугольно-призматическая сетка содержит 26 476 ячеек. В этой задаче количество ячеек достаточно мало, однако расчеты занимают продолжительное время, подавляющая часть которого тратится на моделирование химических взаимодействий. Результаты расчетов продемонстрированы на рис. 6, из которого следует, что присутствие нитрат-иона в более высокой концентрации приводит к худшей сорбции стронция и соответственно к более быстрому его движению в пласте. В рамках настоящей работы нас интересует в большей степени возможность ускорения модели переноса с химическими взаимодействиями с помощью распараллеливания вычислений.

Три выбранные тестовые задачи призваны характеризовать эффективность распараллеливания для моделей разных физических процессов. В первых двух из них основные затраты времени приходятся на сборку и решение распределенных по процессорам систем линейных уравнений большой размерности как в фильтрационной, так и в миграционной подзадачах. В третьей тестовой задаче, напротив, наиболее ресурсоемкими являются расчеты химических взаимодействий, выполняемые локально на ячейках сетки.

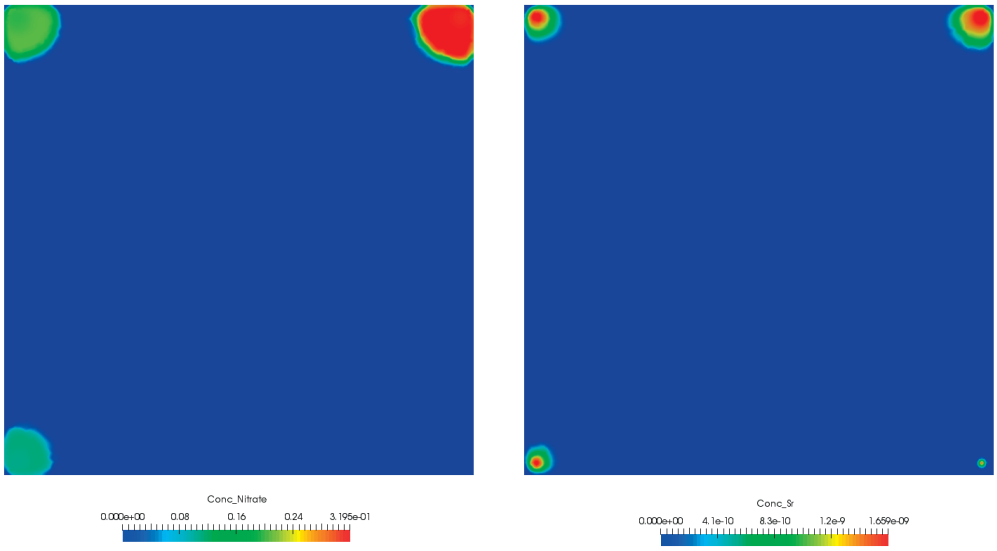


Рис. 6. Распределение нитрат-иона (a) и стронция (b) в расчетной области для модели Pchem на момент времени 300 суток

5. Результаты численных экспериментов.

5.1. Параметры вычислительного кластера. Численные эксперименты проводились на кластере ИВМ РАН [20] на вычислительных узлах сегментов x6core и x12core. Эти узлы имеют следующие характеристики, соответственно, для сегмента x6core:

— compute Node Asus RS704D-E6;

— 12 ядер (два 6-ядерных процессора Intel Xeon X5650@2.67ГГц);

— оперативная память: 24 Гб;

— операционная система: SUSE Linux Enterprise Server 11 SP1 (x86_64)

и для сегмента x12core:

— compute Node Arbyte Alkazar+ R2Q50;

— 24 ядра (два 12-ядерных процессора Intel Xeon E5-2670v3@2.30ГГц);

— оперативная память: 64 Гб;

— операционная система: SUSE Linux Enterprise Server 11 SP3 (x86_64).

5.2. Результаты параллельных расчетов. Распределение сеточных данных по процессорам происходит при помощи общедоступного пакета ParMetis [13], который интегрирован в программную платформу INMOST.

Для решения линейных систем задач фильтрации и переноса использовался интегрированный в платформу INMOST решатель из пакета PETSc [10] с применением аддитивного метода Шварца (`pc_type=asm`) с перекрытием блоков (`pc_asm_overlap=3`), а также использованием неполного треугольного разложения ILU ($k = 3$) (`sub_pc_factor_levels=3`). Итерации BiCGstab (`ksp_type=bcgs`) останавливались при падении невязки в 10^9 раз (`ksp_rtol=1e-9`). При решении задач фильтрации применялось левостороннее предобусловливание (`ksp_pc_side=left`), а для задач переноса, для повышения надежности решения, правостороннее предобусловливание (`ksp_pc_side= right`).

В табл. 1–6 приведены результаты расчетов и использованы следующие обозначения:

- p — количество используемых вычислительных ядер;
- $T = T(p)$ — время решения задачи в секундах на p вычислительных ядрах;
- S_q — ускорение времени расчета на p вычислительных ядрах относительно исходного времени расчета на q вычислительных ядрах: $S_q = T(q)/T(p)$;
- $S_{p/2}$ — ускорение времени расчета на p вычислительных ядрах относительно времени расчета на $p/2$ вычислительных ядрах: $S_{p/2} = T(p/2)/T(p)$.

Таблица 1
Модель Pflow, сегмент x12core

p	T (с.)	S_1	$S_{p/2}$
1	2044.97	1.00	—
2	1086.56	1.88	1.88
4	571.91	3.57	1.89
8	324.19	6.30	1.76
16	205.35	9.95	1.57
32	140.97	14.50	1.45
64	105.25	19.42	1.33
128	46.49	43.98	2.26
192	32.16	63.58	1.44

Таблица 2
Модель Pflow, сегмент x6core

p	T (с.)	S_1	$S_{p/2}$
1	2686.60	1.00	—
2	1541.19	1.74	1.74
4	784.49	3.42	1.96
8	529.28	5.07	1.48
16	484.56	5.54	1.09
32	178.52	15.04	2.71
64	80.46	33.39	2.21
128	43.78	61.36	1.83

Таблица 3
Модель Ptran, сегмент x12core

p	T (с.)	S_1	$S_{p/2}$
2	57027.52	1.00	—
4	30899.76	1.84	1.84
8	17928.51	3.18	1.72
16	10795.76	5.28	1.66
32	7692.88	7.41	1.40
64	4111.9	13.86	1.87
128	2065.94	27.60	1.99
192	1542.59	36.96	1.33

В табл. 1 и 2 приведены результаты расчетов задачи фильтрации для модели Pflow на сегментах x12core и x6core соответственно. Можно отметить, что при использовании одного вычислительного ядра на более современных вычислительных узлах сегмента x12core время решения меньше, чем на узлах x6core, однако на 64 и 128 ядрах ситуация изменяется на противоположную. В целом, приведенные результаты показывают достаточно хорошее ускорение вплоть до 192 вычислительных ядер. Несколько случаев сверхлинейного ускорения $S_{p/2}$ можно объяснить более эффективным использованием кэш-памяти в этих конкретных случаях при уменьшении размерности решаемой подзадачи.

В табл. 3 и 4 показаны результаты расчетов задачи переноса для модели Ptran на сегментах x12core и x6core соответственно. Приведенные данные демонстрируют стабильное ускорение при использовании до 128 и 192 вычислительных ядер. Полученное при этом ускорение для 128 вычислительных ядер относительно времени решения на 2 ядрах составляет 27.60 и 29.25, что соответствует 43% и 45% эффективности расчетов на сегментах x12core и x6core соответственно. Это является достаточно хорошим результатом, несмотря на использование относительно большого перекрытия блоков в 3 слоя ячеек (`pc_asm_overlap=3`), которое было выбрано для повышения надежности решения.

В табл. 5 и 6 даны результаты расчетов для задачи переноса модели Pchem с использованием химических взаимодействий при расчетах на 100 и 300 дней соответственно. Отсутствие в таблицах расчетов на малом числе вычислительных ядер обусловлено системным ограничением на общее время решения задачи в используемой системе организации очередей. Тем не менее, на 128 вычислительных ядрах можно наблюдать эффективность расчетов 45% и 59% относительно времени решения на 2 и 8 ядрах, приведенных в табл. 5 и 6 соответственно.

Важным показателем параллельной эффективности применяемого метода решения систем линейных уравнений является зависимость среднего количества итераций, затрачиваемых на достижение заданной точности решения, от количества вычислительных ядер. Для моделей Pflow и Ptran при увеличении

Таблица 4
Модель Ptran, сегмент x6core

p	T (с.)	S_1	$S_{p/2}$
2	72267.77	1.00	—
4	37872.65	1.90	1.90
8	23490.57	3.07	1.61
16	14593.40	4.95	1.60
32	8639.32	8.36	1.68
64	4377.29	16.50	1.97
128	2470.17	29.25	1.77

Таблица 5
Модель Pchem (100 дней), сегмент x6core

p	T (с.)	S_1	$S_{p/2}$
2	86215.83	1.00	—
4	41767.22	2.06	2.06
8	25631.87	3.36	1.62
16	15313.55	5.63	1.67
32	9283.06	9.28	1.64
64	4697.10	18.35	1.97
128	2982.72	28.90	1.57

Таблица 6
Модель Pchem (300 дней), сегмент x6core

p	T (с.)	S_1	$S_{p/2}$
8	85080.86	1.00	—
16	48762.31	1.74	1.74
32	28159.37	3.02	1.77
64	14993.77	5.67	1.87
128	9004.06	9.44	1.60

количества ядер от 2 до 128 рост количества линейных итераций составил всего 16% и 10% соответственно. Для модели Pchem при расчете на 100 дней для того же количества вычислительных ядер рост числа линейных итераций составил 11%. Такое незначительное увеличение количества итераций характеризует параллельную эффективность и надежность применяемого метода решения линейных систем.

Суммируя результаты, приведенные в табл. 1–6, на рис. 7 мы привели графики уменьшения времени расчета для кода GeRa для всех рассмотренных моделей при использовании различного количества вычислительных ядер сегментов x6core и x12core. Приведенные данные подтверждают факт монотонного уменьшения времени расчета с ростом количества используемых ядер.

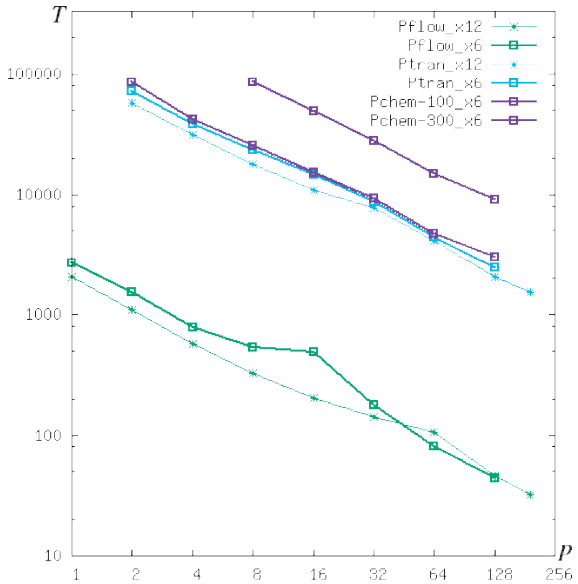


Рис. 7. Уменьшение времени расчета T для кода GeRa при использовании различного количества вычислительных ядер p

6. Заключение. В заключение данной работы сформулируем основные полученные результаты:

— проведены расчеты вплоть до 198 вычислительных ядер на кластере ИВМ РАН;

— для всех приведенных расчетов получено разумное ускорение, например, на 128 вычислительных ядрах эффективность параллельных вычислений составила около 50%;

— наибольшая эффективность параллельных вычислений получена для задач переноса с учетом химических взаимодействий;

— для всех приведенных расчетов наблюдался монотонный рост ускорения при увеличении числа используемых вычислительных ядер.

Применение распараллеливания вычислений позволит пользователям не только сократить время расчета для конкретной модели, но и за выбранное референтное время получить решение с большей точностью при построении более подробной расчетной сетки. Это позволит повысить точность и надежность моделирования, а также, в конечном счете, и достоверность результатов, полученных с помощью расчетного кода GeRa.

СПИСОК ЛИТЕРАТУРЫ

1. *Капырин И.В., Иванов В.А., Копытов Г.В., Уткин С.С.* Интегральный код GeRa для обоснования безопасности захоронения РАО // Горный журнал. 2015. № 10. 44–50.
2. *Freedman V.L., Chen X., Finsterle S., et al.* A high-performance workflow system for subsurface simulation // Environmental Modelling & Software. 2014. **55**. 176–189.
3. *Hammond G.E., Lichtner P.C., Mills R.T.* Evaluating the performance of parallel subsurface simulators: an illustrative example with PFLOTRAN // Water Resources Research. 2014. **50**, N 1. 208–228.

4. *Kolditz O., Bauer S., Bilke L., et al.* OpenGeoSys: an open-source initiative for numerical simulation of thermo-hydro-mechanical/chemical (THM/C) processes in porous media // Environmental Earth Sciences. 2012. **67**, N 2. 589–599.
5. *Плёнкин А.В., Чернышенко А.Ю., Чугунов В.Н., Капырин И.В.* Методы построения адаптивных неструктурированных сеток для решения гидрогеологических задач // Вычислительные методы и программирование. 2015. **16**. 518–533.
6. PHREEQC (Version 3) — a computer program for speciation, batch-reaction, one-dimensional transport, and inverse geochemical calculations. https://wwwbrr.cr.usgs.gov/projects/GWC_coupled/phreeqc.
7. *Charlton S.R., Parkhurst D.L.* Modules based on the geochemical model PHREEQC for use in scripting and programming languages // Computers & Geosciences. 2011. **37**, N 10. 1653–1663.
8. *Василевский Ю.В., Коньшин И.Н., Копытов Г.В., Терехов К.М.* INMOST — программная платформа и графическая среда для разработки параллельных численных моделей на сетках общего вида. М.: Изд-во Моск. ун-та, 2013.
9. *Danilov A.A., Terekhov K.M., Konshin I.N., Vassilevski Y.V.* Parallel software platform INMOST: a framework for numerical modeling // Supercomputing Frontiers and Innovations. 2015. **2**, N 4. 55–66.
10. PETSc — a suite of data structures and routines for the scalable (parallel) solution of scientific applications modeled by partial differential equations. http://www.mcs.anl.gov/petsc.
11. ParaView — an open-source, multi-platform data analysis and visualization application. https://www.paraview.org.
12. Qt — Cross-platform software development for embedded & desktop. https://www.qt.io.
13. ParMETIS — Parallel graph partitioning and fill-reducing matrix ordering. http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview.
14. VTK — The Visualization Toolkit. https://www.vtk.org.
15. SVN — an open source version control system. https://subversion.apache.org.
16. *Konshin I., Kapyrin I., Nikitin K., Terekhov K.* Application of the parallel INMOST platform to subsurface flow and transport modelling // Lecture Notes in Computer Science. Vol. 9574. Heidelberg: Springer, 2016. 277–286.
17. *Konshin I., Kapyrin I.* Scalable computations of GeRa code on the base of software platform INMOST // Lecture Notes in Computer Science. Vol. 10421. Heidelberg: Springer, 2017. 433–445.
18. *Рыбальченко А.И., Пименов М.К., Костин П.П. и др.* Глубинное захоронение жидких радиоактивных отходов. М.: ИздАТ, 1994.
19. *Болдырев К.А., Капырин И.В., Константинова Л.И., Захарова Е.В.* О моделировании сорбции стронция на породах в условиях высокой засоленности раствора нитратом натрия // Радиохимия. 2016. **58**, № 3. 211–217.
20. Кластер ИВМ РАН. http://cluster2.inm.ras.ru.

Поступила в редакцию
19.07.2018

Parallel Computations in the Hydrogeological Computational Code GeRa: Organization and Efficiency

I. V. Kapyrin [1], I. N. Konshin [2], G. V. Kopytov [3], and V. K. Kramarenko [4]

[1] *Nuclear Safety Institute, Russian Academy of Sciences; ulitsa Bol'shaya Tul'skaya 52, Moscow, 115191, Russia; Ph.D., Head of Laboratory, e-mail: ivan.kapyrin@gmail.com*

[2] *Dorodnitsyn Computing Center, Russian Academy of Sciences; ulitsa Vavilova 40, Moscow, 119333, Russia; Ph.D., Scientist, e-mail: igor.konshin@gmail.com*

[3] *Immanuel Kant Baltic Federal University, Institute of Physics, Mathematics and Information Technology; ulitsa A. Nevskogo 14, Kaliningrad, 236016, Russia; Ph.D., Associate Professor, e-mail: gkopytov@yandex.ru*

[4] *Marchuk Institute of Numerical Mathematics, Russian Academy of Sciences; ulitsa Gubkina 8, Moscow, 119333, Russia; Junior Scientist, e-mail: kramarenko.vasiliy@gmail.com*

Received July 19, 2018

Abstract: The structure, functionality and parallel computation organization of the computational code GeRa designed for hydrogeological modeling are considered. The principles of multiprocessor task launching on a remote cluster directly from the GeRa graphical user interface are described. The results of parallel computations for three representative models are presented and analyzed from the standpoint of parallel efficiency.

Keywords: groundwater flow, transport in geological media, computational code, parallel computations, speed-up.

References

1. I. V. Kapyrin, V. A. Ivanov, G. V. Kopytov, and S. S. Utkin, "Integral Code GeRa for Radioactive Waste Disposal Safety Validation," Gornyi Zh., No. 10, 44–50 (2015).

2. V. L. Freedman, X. Chen, S. Finsterle, et al., "A High-Performance Workflow System for Subsurface Simulation," Environ. Model. Softw. **55**, 176–189 (2014).

3. G. E. Hammond, P. C. Lichtner, and R. T. Mills, "Evaluating the Performance of Parallel Subsurface Simulators: An Illustrative Example with PFLOTRAN," Water Resour. Res. **50**, 208–228 (2014).

4. O. Kolditz, S. Bauer, L. Bilke, et al., "OpenGeoSys: An Open-Source Initiative for Numerical Simulation of Thermo-Hydro-Mechanical/Chemical (THM/C) Processes in Porous Media," Environ. Earth Sci. **67** (2), 589–599 (2012).

5. A. V. Plenkin, A. Yu. Chernyshenko, V. N. Chugunov, and I. V. Kapyrin, "Adaptive Unstructured Mesh Generation Methods for Hydrogeological Problems," Vychisl. Metody Programm. **16**, 518–533 (2015).

6. PHREEQC (Version 3) — A Computer Program for Speciation, Batch-Reaction, One-Dimensional Transport, and Inverse Geochemical Calculations. https://wwwbrr.cr.usgs.gov/projects/GWC_coupled/phreeqc. Cited August 22, 2018.

7. S. R. Charlton and D. L. Parkhurst, "Modules Based on the Geochemical Model PHREEQC for Use in Scripting and Programming Languages," Comput. Geosci. **37** (10), 1653–1663 (2011).

8. Yu. V. Vassilevski, I. N. Konshin, G. V. Kopytov, and K. M. Terekhov, *INMOST — Programming Platform and Graphical Environment for Development of Parallel Numerical Models on Various Grids* (Mosk. Gos. Univ., Moscow, 2013) [in Russian].

9. A. A. Danilov, K. M. Terekhov, I. N. Konshin, and Yu. V. Vassilevski, "Parallel Software Platform INMOST: A Framework for Numerical Modeling," Supercomput. Front. Innov. **2** (4), 55–66 (2015).

10. PETSc — A Suite of Data Structures and Routines for the Scalable (Parallel) Solution of Scientific Applications Modeled by Partial Differential Equations. http://www.mcs.anl.gov/petsc. Cited August 22, 2018.

11. ParaView — An Open-Source, Multi-Platform Data Analysis and Visualization Application. https://www.paraview.org. Cited August 22, 2018.

12. Qt — Cross-Platform Software Development for Embedded & Desktop. https://www.qt.io. Cited August 22, 2018.

13. ParMETIS — Parallel Graph Partitioning and Fill-Reducing Matrix Ordering. http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview. Cited August 22, 2018.

14. VTK — The Visualization Toolkit. https://www.vtk.org. Cited August 22, 2018.

15. SVN – An Open Source Version Control System. https://subversion.apache.org. Cited August 22, 2018.

16. I. Konshin, I. Kapyrin, K. Nikitin, and K. Terekhov, "Application of the Parallel INMOST Platform to Subsurface Flow and Transport Modelling," in *Lecture Notes in Computer Science* (Springer, Heidelberg, 2016), Vol. 9574, pp. 277–286.

17. I. Konshin and I. Kapyrin, "Scalable Computations of GeRa Code on the Base of Software Platform INMOST," in *Lecture Notes in Computer Science* (Springer, Heidelberg, 2017), Vol. 10421, pp. 433–445.

18. A. I. Rybal'chenko, M. K. Pimenov, P. P. Kostin, et al., *Underground Disposal of Liquid Radioactive Wastes* (IzdAT, Moscow, 1994) [in Russian].

19. K. A. Boldyrev, I. V. Kapyrin, L. I. Konstantinova, and E. V. Zakharova, "Simulation of Strontium Sorption onto Rocks at High Concentrations of Sodium Nitrate in the Solution," Radiokhimiya **58** (3), 211–217 (2016) [Radiochemistry **58** (3), 243–251 (2016)].

20. Cluster at the Institute of Numerical Mathematics, Russian Academy of Sciences. http://cluster2.inm.ras.ru. Cited August 22, 2018.