

ВСЕРОССИЙСКАЯ ЛЕТНЯЯ ХLI
МОЛОДЕЖНАЯ ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 3–14 июля 2023 года)

.....

ЛЕТНЯЯ ХLIII ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 7–12 июля 2024 года)

ТЕЗИСЫ ДОКЛАДОВ

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ
И МАТЕМАТИЧЕСКОЙ ГЕОФИЗИКИ СО РАН
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ВСЕРОССИЙСКАЯ ЛЕТНЯЯ ХLI МОЛОДЕЖНАЯ
ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 3–14 июля 2023 года)

•

ЛЕТНЯЯ ХLIII ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 1–12 июля 2024 года)

Тезисы докладов

Новосибирск
2026

УДК 004.4
ББК 32.973
В85

В85 Всероссийская летняя ХLI молодежная Школа-конференция по параллельному программированию (г. Новосибирск, 3–14 июля 2023 г.). Летняя ХLIII школа-конференция по параллельному программированию (г. Новосибирск, 1–12 июля 2024 г.): Тез. докл. / отв. ред. д-р техн. наук, проф. В. Э. Малышкин ; Новосиб. гос. ун-т. — Новосибирск : ИПЦ НГУ, 2026. — 32 с.

ISBN 978-5-4437-1808-8

В сборнике представлены тезисы докладов участников двух летних Школ-конференций по параллельному программированию, которые были организованы ИВМиМГ СО РАН, НГУ и НГТУ и проходили в Новосибирском государственном университете с 3 по 14 июля 2023 года и с 1 по 12 июля 2024 года.

Для студентов, магистрантов, аспирантов и молодых ученых, использующих высокопроизводительные вычислительные ресурсы в научной и учебной деятельности.

УДК 004.4
ББК 32.973

© Институт вычислительной
математики и математической
геофизики СО РАН, 2026

© Новосибирский государственный
университет, 2026

ISBN 978-5-4437-1808-8

DOI 10.25205/978-5-4437-1808-8

© Новосибирский государственный
технический университет, 2026

ПРЕДИСЛОВИЕ

Организаторами Всероссийской XLI летней молодежной Школы-конференции по параллельному программированию с международным участием (URL: <https://ssd.sccc.ru/ru/school/2023s>) и Летней XLIII школы-конференции по параллельному программированию (URL: <https://ssd.sccc.ru/ru/school/2024s>) являлись Институт вычислительной математики и математической геофизики СО РАН (ИВМиМГ СО РАН), Новосибирский национальный исследовательский государственный университет (НГУ) и Новосибирский государственный технический университет (НГТУ).

Основная цель школы-конференции — дать возможность студентам, магистрантам, аспирантам и молодым ученым продемонстрировать собственные результаты и разработки, обсудить актуальные фундаментальные научные проблемы в области создания технологий параллельного программирования и их применения для решения практических задач, сформировать и укрепить научные контакты. В форме проектной работы участники имеют возможность изучить современные технологии организации высокопроизводительных вычислений, получить опыт исследовательской работы. Участие в проектной работе позволяет студентам сориентироваться в выборе научного руководителя и темы выпускных квалификационных работ. В рамках программы школы-конференции, сотрудники ИВМиМГ СО РАН, НГУ, НГТУ прочитали лекции, представив обзор современной проблематики в сфере технологий параллельного программирования и тенденций их развития.

Школа-конференция является частью систематической работы организаторов по развитию научной коммуникации и научных кадров в области проблем параллельного программирования. В сотрудничестве с партнерами раз в два года, начиная с 1991 г., проводится Международная научная конференция «Parallel Computing Technologies» (PaCT, <http://ssd.sccc.ru/en/conference>). Ежегодно с 2000 г. ИВМиМГ СО РАН, НГУ и НГТУ организуют зимнюю и летнюю школы по параллельному программированию для проходящих в науку студентов, с 2017 года летняя школа трансформировалась в школу-конференцию, программа которой составляется из лекций ведущих ученых,

двухнедельных образовательно-научных проектов и конференционной части с докладами участников о результатах предложенной организаторами проектной работы и собственных научных исследований в период до конференции.

Школа-конференция специализируется в области научных проблем, связанных с развитием технологий решения задач на параллельных вычислительных системах. Приложения параллельного программирования в таком контексте рассматриваются постольку, поскольку позволяют сформулировать требования к развитию технологий программирования, организации управления данными и вычислениями и оценить технологии с точки зрения применимости к реальным задачам.

Высокопроизводительные вычислительные системы дают возможность проводить крупномасштабное численное моделирование и обработку большого объема данных, обеспечивая развитие науки, решение практических задач промышленности и государственного управления. Вместе с тем с учетом современного состояния технологий организации вычислений на суперкомпьютерах разработчику приложений нужно владеть теорией и практикой системного параллельного программирования, быть способным разрабатывать корректные параллельные алгоритмы и применять низкоуровневые средства программирования, интерфейсы управления данными и вычислительными задачами. По этой причине затруднено эффективное применение высокопроизводительных вычислительных систем прикладными специалистами, сдерживается развитие науки и технологий.

Школы-конференции ИВМиМГ СО РАН, НГУ, НГТУ по параллельному программированию способствуют продвижению исследований в области актуальной проблемы создания высокоуровневых инструментов параллельного программирования, в частности предметно-ориентированных систем программирования, систем для автоматизированного накопления и переиспользования алгоритмов и программ, параллельных алгоритмов для решения конкретных прикладных задач, повышение эффективности работы вычислительных центров.

ПРОГРАММА ЛЕКЦИЙ ВСЕРОССИЙСКОЙ ЛЕТНЕЙ ХЛІ МОЛОДЕЖНОЙ ШКОЛЫ-КОНФЕРЕНЦИИ ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

Об автоматическом синтезе параллельных программ на основе баз активных знаний

*Перепёлкин Владислав Александрович,
канд. техн. наук, науч. сотр. ИВМиМГ СО РАН*

Существует фундаментальная причина, из-за которой существует такое явление, как программирование, из-за которой компьютер не может сам, непосредственно, делать то, что человек от него хочет, без необходимости человеку формулировать задачу для компьютера в виде компьютерной программы. Тем не менее, компьютер может очень многое взять на себя и в самом процессе программирования. Где же заканчиваются принципиальные возможности компьютера и начинается эксклюзивная область деятельности человека? Ответ на этот вопрос позволяет выстраивать систему программирования так, чтобы снять с человека те заботы, которые может взять на себя компьютер — ни больше, ни меньше. В лекции этот вопрос рассматривается с позиции концепции активных знаний. Также в лекции рассматривается, как может быть создана система автоматического конструирования параллельных программ на основе баз активных знаний — специальной конструкции, позволяющей описать сразу множество алгоритмов и программ предметной области в такой форме, что наиболее подходящий алгоритм и программа решения поставленной задачи будут сконструированы автоматически.

Методы отладки параллельных программ

*Власенко Андрей Юрьевич,
канд. техн. наук, науч. сотр. ИВМиМГ СО РАН*

Трудозатраты на отладку программ в общем цикле разработки нередко составляют более 50 %. Для параллельных программ эта доля зачастую еще больше. Причина этого в новых по сравнению с «последовательным» программированием ошибках, обусловленных взаимодействием параллельно работающих процессов/потоков и ограничениях, накладываемых используемыми средствами программирования. Так, даже поверхностный анализ стандарта MPI первой версии дает понимание того, что пользователь может столкнуться как минимум с несколькими десятками ошибок новых типов.

В лекции приводятся наиболее общие классы ошибок, возникновение которых не зависит от языка и применяемой технологии параллельного программирования (дедлоки, гонки данных, ливлоки, «голодание» и др.) Ошибочные ситуации демонстрируются на примерах «из жизни». В следующей части идет описание различных методов отладки параллельных программ (как «ручных», так и автоматизированных). Приводится их сравнительный анализ и примеры инструментальных средств, использующих данные методы.

На лекции предусмотрена демонстрация работы MPI-программы под управлением диалогового отладчика gdb.

Моделирование пространственной динамики клеточными автоматами

*Медведев Юрий Геннадьевич,
канд. техн. наук, ст. науч. сотр. ИВМиМГ СО РАН*

Клеточно-автоматное моделирование является мощным инструментом для исследования поведения сложных систем. Лекция состоит из трех частей. В первой части рассказывается, что

такое клеточный автомат. В качестве примера используется игра «Жизнь» Дж. Конвея. Во второй части дается представление о том, как клеточный автомат используется в качестве имитационной модели пространственной динамики. Рассматриваются модели двумерной и трехмерной диффузии/теплопроводности, ламинарных и турбулентных потоков жидкостей и газов, просачивания жидкости через пористую среду, распространения ударной волны, затрагиваются вопросы динамических граничных условий. Третья часть лекции посвящена методам программной реализации клеточных автоматов. Описываются способы последовательной и параллельной программной реализации, рассказывается о методах балансировки нагрузки вычислительного кластера. Отдельное внимание уделяется методам реализации, использующим библиотеку клеточно-автоматных топологий CATlib, которая позволяет автоматизировать ряд этапов разработки. Лекция рассчитана на студентов инженерно-физических и математических специальностей вузов.

Параллельные методы декомпозиции областей с графовыми предобуславливателями для решения многомерных краевых задач

*Ильин Валерий Павлович,
д-р физ.-мат. наук, гл. науч. сотр. ИВМиМГ СО РАН*

Рассматриваются подход к декомпозиции области моделирования с определением разделяющей подобласти, образующей макросетку и состоящей из интерфейсных узлов для всех подобластей. Обращение соответствующей алгебраической подсистемы осуществляется с помощью экономичных параллельных алгоритмов, а общий итерационный процесс формируется на основе блочных методов неполной факторизации в подпространствах Крылова. Рассмотрение алгоритмов ведется на примерах двумерных и трехмерных регулярных сеток с формированием СЛАУ достаточно простой структуры.

Gurieva Y.L., Il'in V.P., Kozlov D.I. Parallel Domain Decomposition Methods with Graph Preconditioning // Параллельные вычислительные технологии — XVII всероссийская научная конференция с международным участием, ПАВТ'2023, г. Санкт-Петербург, 28–30 марта 2023 г. Короткие статьи и описания плакатов. Челябинск: Изд. центр ЮУрГУ, 2023. С. 215–228.

ПРОГРАММА ЛЕКЦИЙ ЛЕТНЕЙ XLIII ШКОЛЫ-КОНФЕРЕНЦИИ ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

База активных знаний как практичный способ представления множества программ предметной области

*Переёлкин Владислав Александрович,
канд. техн. наук, ст. науч. сотр. ИВМиМГ СО РАН*

Сложность программирования во многом обусловлена тем, что разработка достаточно хорошей для практического использования программы требует, вообще говоря, учета особенностей конкретной предметной области, к которой она относится. Универсальных подходов, которые обеспечили бы создание таких программ в любой предметной области, не существует вследствие фундаментальных причин, раскрытых в теории алгоритмов. Концепция активных знаний — это методология автоматизации конструирования программ путем создания особого описания конкретной предметной области, которое позволяет автоматически конструировать множество программ в этой предметной области. В лекции излагаются основные принципы, по которым строятся базы активных знаний, а также то, как они используются для автоматического конструирования программ.

Основы клеточно-автоматного моделирования природных процессов

*Медведев Юрий Геннадьевич,
канд. техн. наук, ст. науч. сотр. ИВМиМГ СО РАН*

В лекции рассмотрены основные принципы клеточно-автоматного моделирования, как одного из мощных инструментов для изучения

и анализа природных процессов. Клеточные автоматы представляют собой дискретные модели, состоящие из массива клеток, каждая из которых может находиться в одном из конечного числа состояний. В лекции идет речь о том, как эти модели могут быть использованы для моделирования различных природных явлений, таких как газовые потоки, диффузионный процесс и т.п. В лекции уделено внимание основным компонентам клеточных автоматов, включая функции переходов клеток в новые состояния, начальные и граничные условия. Также рассмотрены примеры успешного применения клеточно-автоматного подхода в физике, химии и других науках. Лекция направлена на формирование у слушателей понимания возможностей и ограничений клеточно-автоматного моделирования, а также на развитие навыков критического анализа результатов моделирования.

Подходы к высокоуровневой организации расчетов на высокопроизводительных вычислительных системах

*Городничев Максим Александрович,
науч. сотр. ИВМиМГ СО РАН*

Рассматривается проблема комплексной организации решения задач численного моделирования и анализа данных с обеспечением выполнения расчетов на высокопроизводительных вычислительных системах (ВВС). Обсуждаются организационные особенности проведения расчетов на суперкомпьютерах. Рассматриваются категории пользователей суперкомпьютеров. Проводится анализ требований к автоматизации проведения расчетов, управлению данными вычислительных экспериментов. Дается исторический обзор в области организации вычислений на суперкомпьютерах, грид и облачных системах, обзор существующих подходов к повышению уровня взаимодействия пользователя с суперкомпьютерами, рассматриваются существующие программные инструменты: универсальные вычислительные платформы, комплексные предметно-ориентированные пакеты, системы управления выполнением вычислительных сцена-

риев (workflow), предметно-ориентированные языки программирования, системы управления научными данными.

Современные методы моделирования химической кинетики в мультифизичных задачах

Пененко Алексей Владимирович,

д-р физ.-мат. наук, зам. дир. по научной работе ИВМиМГ СО РАН

В лекции рассказывается об основных современных подходах к компьютерному моделированию кинетики химических реакций, возникающих в промышленно-ориентированных приложениях. Рассматривается проблематика моделирования в различных классах задач с учетом особенностей моделируемых явлений. Дается анализ сложностей решения задач химической кинетики и обсуждается математический аппарат для построения моделей, включая моделирование на основе ОДУ и нейросетевые модели. Вычислительная сложность задач требует поиска вариантов ускорения расчетов. Рассматриваются методы редукции моделей химической кинетики. Дается обзор опыта ИВМиМГ СО РАН в решении задач химической кинетики, формулируется стратегическая задача создания отечественного программного обеспечения в области химико-кинетических расчетов.

От компиляции к генерации параллельных программ

Мальшкин Виктор Эммануилович,

гл. науч. сотр., зав. лаб. ИВМиМГ СО РАН

Компиляция как способ получения исполняемой программы из исходных кодов на языке программирования высокого уровня является наиболее распространенным в мире. Тем не менее, он имеет свои ограничения, связанные с тем, что компилятор конструирует программу из стандартных блоков, что неизбежно вызывает «ме-

жмодульное трение», которое негативно сказывается на нефункциональных свойствах программы. Альтернативным является генерация — подход, при котором программа строится автоматически из машинных команд на основе базы знаний предметной области, представленной в форме сети операций и переменных. Этот подход требует дополнительных ресурсов для вывода достаточно хорошего решения, что вполне приемлемо при использовании современной вычислительной техники.

Особенности разработки и реализации средств автоматического конструирования параллельных программ

*Беляев Николай Алексеевич,
Exasol AG*

Качество реализации систем автоматического конструирования параллельных программ оказывает непосредственное влияние на производительность конструируемых параллельных программ. В лекции рассматриваются широко известные техники и приемы системного параллельного программирования, которые применяются при реализации основных компонент систем автоматического конструирования параллельных программ (таких, как коммуникационная подсистема, модуль распределенного исполнения графов задач и пр.). В частности, предлагаются приемы проектирования архитектуры исполнительных систем для систем автоматического конструирования параллельных программ и примеры реализации распределенных очередей задач. Приводятся примеры применения рассмотренных техник и приемов при реализации системы автоматического конструирования параллельных программ LuNA.

ТЕЗИСЫ ДОКЛАДОВ УЧАСТНИКОВ

Разработка программной системы клеточно-автоматного моделирования газовых потоков

Е. К. Бурнышев

Новосибирский государственный университет

✉ e.burnyshev@g.nsu.ru

Современная наука предлагает большое количество подходов к моделированию естественных процессов. Одним из перспективных методов моделирования пространственно-динамических процессов является клеточно-автоматное моделирование [1]. Данный подход, в частности, используется для описания систем, когда моделируемое явление представлено нелинейными или разрывными функциями.

Широкое применение клеточно-автоматный подход получил при моделировании явлений газовой динамики. Высокая степень соответствия результатов имитации газодинамических процессов при помощи клеточных автоматов аналитическим решениям, а также свойство естественного параллелизма клеточного автомата являются его главными достоинствами. Благодаря этим фактам ученые все чаще прибегают к использованию клеточно-автоматных моделей в своих исследованиях.

Работа посвящена созданию программной системы клеточно-автоматного моделирования газовых потоков. В ходе проведения исследования разработаны: алгоритм перевода растрового изображения в гексагональную сетку клеточно-автоматной модели FHP-MP [2], алгоритм конвертации геометрии конечно-элементной модели пакета ЛОГОС Аэро-гидро в клеточный автомат FHP-MP. Алгоритмы были реализованы в виде отдельных программных модулей препроцессора и конвертера.

Препроцессорный программный модуль отвечает за преобразование растрового изображения, содержащего геометрию расчетной

области, во входной файл симулятора модели FHP-MP. Модуль конвертера позволяет преобразовать геометрию области моделирования пакета ЛОГОС Аэро-Гидро в клеточный автомат FHP-MP. Разработка велась на языке C++ с использованием библиотеки клеточно-автоматных топологий CATlib [3].

В рамках работы разработаны и реализованы два новых алгоритма, которые легли в основу препроцессора и конвертера. Проведено тестирование разработанных модулей и системы моделирования в целом на нескольких базовых задачах газовой динамики. Наблюдалось качественное сходство полученных в ходе вычислительного эксперимента решений с численным эталоном.

Литература

1. Бандман О.Л. Клеточно-автоматные модели пространственной динамики // Системная информатика. 2006. № 10. С. 59–113.
2. Медведев Ю.Г. Многочастичная клеточно-автоматная модель потока жидкости FHP-MP // Вестник Томского государственного университета. 2009. № 1 (6). С. 33–40.
3. Medvedev Yu. G. Architecture of the cellular automata topologies library // Bull. Nov. Comput. Center Comput. Sci. 2022. Vol. 46. P. 27–41.

Научный руководитель — канд. техн. наук Ю.Г. Медведев,
ИВМиМГ СО РАН, НГУ, НГТУ

Профилерование библиотеки клеточно-автоматных топологий для синхронных клеточных автоматов на мультикомпьютере

К. А. Глинский

Сибирский государственный университет телекоммуникаций
и информатики, г. Новосибирск

✉ kirill_glinskij@mail.ru

Важную часть разработки программного обеспечения занимает их тестирование. Тестирование библиотек позволяет убедиться в их правильной работе, выявить и исправить возможные ошибки и дефекты. Кроме того, тестирование позволяет оценить производительность библиотеки, что является важным аспектом ее успешной интеграции и использования.

Работа посвящена анализу производительности библиотеки клеточно-автоматных топологий [1] с помощью средств профилирования, в частности программы ITAS. Библиотека клеточно-автоматных топологий помогает пользователю создавать средства моделирования различных процессов и может работать на машинах с распределенной памятью, используя технологию MPI.

Тестирование проводилось с целью определить долю накладных расходов на MPI-коммуникации и оценить целесообразность оптимизации библиотеки.

Работу программы-симулятора можно разделить на четыре части.

1. Чтение и рассылка файла с начальными данными.
2. Итерации клеточного автомата.
3. Сборка и сохранение результата в файл.
4. Физическая запись файла с результатом.

Процент накладных расходов на коммуникации определялся как раз во второй части, когда все процессы получили данные для расчетов, производят расчеты, обмениваясь данными на каждой итерации, и еще не начали передавать данные основному процессу. Тестирование проводилось на двух кластерах. На кластере НГУ из аудиторных компьюте-

ров, с максимальным количеством процессов до 32. А также на кластере МВС-10П МСЦ РАН [2], где проверяли с максимумом в 256 процесса.

Тестирование проводилось на клеточном автомате разделения фаз с бинарным алфавитом [3] в синхронном режиме, в котором функция переходов имеет минимум вычислений. Количество итераций равнялось 1000, а размер массива 20 000 на 20 000 клеток. На кластере НГУ процент накладных расходов составил от 0.2 % до 0.7 %, что является хорошим показателем. На кластере МСЦ накладные расходы, в некоторых случаях, оказались сопоставимы с теми, что и на кластере НГУ, а именно <1 %, но в большинстве случаев от 1 % до 5 %. В редких случаях накладные расходы достигали 30%. Было выявлено, что такие большие задержки вызваны одним процессом, который работал существенно медленнее остальных, чем вызывал ожидание соседних процессов, что, в свою очередь, вызывала каскадные задержки у всех процессов. Так как на кластере для каждого запуска программы могут выделяться разные процессорные ядра доля накладных расходов от запуска к запуску может сильно различаться. Для решения такой проблемы необходимо балансировать нагрузку даже на однородных задачах.

Доля накладных расходов на MPI-коммуникации составила менее 1% либо несколько процентов в случае неоднородности вычислителя. Основываясь на результатах тестов, мы пришли к выводу, что оптимизировать пересылки библиотеки не нужно, но следует реализовать балансировку нагрузки.

Литература

1. Medvedev Yu. G. Architecture of the Cellular-Automata Topologies Library // Bull. Nov. Comp. Center, Comp. Science. 2022. Iss. 46. P. 27–41.
2. Savin G. I., Shabanov B. M., Telegin P. N. et al. Joint supercomputer center of the Russian Academy of Sciences: present and future // Lobachevskii J. Math. 40, 1853–1862 (2019). DOI: 10.1134/S1995080219110271.
3. Бандман О. Л. Клеточно-автоматные модели пространственной динамики // Системная информатика: Сб. науч. тр. Вып. 10. Новосибирск: Изд-во СО РАН, 2006. С. 59–113.

Научный руководитель — канд. техн. наук Ю. Г. Медведев,
ИВМиМГ СО РАН, НГУ, НГТУ

Разработка модулей детекции микросейсмических событий для суперкомпьютерного приложения «Микросейсмический мониторинг»

Ю. Ю. Голыжбина

Новосибирский государственный университет

✉ y.golyzhbina@g.nsu.ru

Одним из наиболее эффективных методов дистанционного контроля при подземной разработке полезных ископаемых является микросейсмический мониторинг, в основе которого лежит принцип регистрации сигналов, возникающих в горных породах [1]. Данные микросейсмического мониторинга характеризуются большим объемом, что приводит к необходимости использовать для их обработки суперкомпьютеры и специальное программное обеспечение.

Цель данной работы — разработка библиотеки модулей для суперкомпьютерного приложения «Микросейсмический мониторинг». Среди восьми разработанных модулей можно выделить три основных.

Первый из них — модуль чтения одномерной скоростной модели. Для обработки файлов, содержащих входные данные разработан модуль разбора файла формата .docx и парсер файла формата .las, необходимые для получения массива скоростей.

Вторым важным модулем является модуль, рассчитывающий время пробега волны, созданной источником сейсмического события, до каждого сейсмоприемника. Данный модуль принимает на вход скоростную модель, координату предполагаемого источника и координаты сейсмоприемников. Для уменьшения времени расчета, заметим, что задача является осесимметричной. Таким образом, время пробега зависит только от глубины и горизонтального расстояния от источника до приемника. Тогда расчет времен сводится к единственному численному решению двумерного уравнения эйконала (вместо трехмерного) для каждой глубины.

Последний из основных модулей — модуль, производящий операцию когерентного суммирования. На вход подаются сейсмотрас-

сы, предварительно прочитанные с помощью парсера .sgy файлов и обработанные функциями, уменьшающими шум, и времена пробега волны от каждого источника до каждого приемника. Когерентное суммирование предполагает суммирование значений амплитуд сейсмограмм по годографу прямой волны из пробного источника [3], это реализуется следующим образом: выбирается начальное время, к которому прибавляется время пробега волны от источника до приемника, к сумме прибавляется значение амплитуды сейсмограммы, соответствующего приемника в соответствующий момент времени. Суммируя таким образом по всем источникам и для каждого начального времени, на выходе получаем куб когерентности, осями которого являются оси x , y , t , где x , y — оси координат источника, t — ось времен возникновения волны.

Модули были написаны на языке Python. Для оптимизации операции когерентного суммирования было разработано C++ расширение для Python. В качестве входных данных в расширении принимаются времена пробега волны от каждого источника до каждого приемника и сейсмотрассы и производится основной цикл суммирования.

Разработанные модули были протестированы на реальных входных данных. Местоположение источника определяется с ошибкой по координате x в 50 метров, по координате y — в 170 метров.

Литература

1. Яскевич С. В., Гречка В. Ю., Дучков А. А. Обработка данных микро-сейсмического мониторинга геодинамических событий с учетом сейсмической анизотропии массива горных пород // Физико-технические проблемы разработки полезных ископаемых. 2014. Вып. 6. С. 41–52.
2. Библиотека Python для решения уравнения эйконала. URL: <https://pypi.org/project/eikonalfm/>
3. Городничев М. А., Дучков А. А., Сарычев В. Г. Эффективная реализация метода когерентного суммирования на ускорителях GPU NVIDIA // Вестник Уфимского государственного авиационного технического университета. 2016. Т. 20, № 1 (71). С. 151–160.

Научные руководители — М. А. Городничев, ИВМиМГ СО РАН, НГУ,
НГТУ, А. С. Матвеев, ИНГГ СО РАН, НГУ

Разработка системы моделирования организма «Физарум полицефалум»

Д. С. Ерошенко

Новосибирский государственный университет

✉ d.eroshenko1@g.nsu.ru

Одноклеточный гриб «Физарум полицефалум» [1] (далее — гриб) является модельным организмом и представляет интерес в научном мире, так как при своей простой одноклеточной структуре он обладает различными свойствами, например, — нахождение кратчайших путей (учитывается в рассматриваемой модели) и способность к запоминанию и обучению (не учитывается в рассматриваемой модели). Поэтому в рамках летней школы была сформирована задача разработать систему, позволяющую промоделировать поведение данного организма при помощи клеточного автомата.

Основой для рассматриваемой системы послужила библиотека CAT, — Cellular Automata Topology [2], — созданная в ИВМиМГ, позволяющая заниматься разработкой клеточно-автоматных моделей, используя высокоуровневые интерфейсы. В рамках настоящей работы были созданы три программных модуля: препроцессор, постпроцессор и симулятор. Кроме того, для удобства задания различных параметров симуляции, был разработан простейший язык-нотация для описания параметров, поддерживающий комментарии, а также интерпретатор этого языка, позволяющий по имени исходного файла и названию параметра получить этот параметр в виде строкового значения.

Была сформирована структура клетки: в ней содержится информация о количестве биомассы гриба (далее — биомасса), количестве сахара и об интенсивности запаха сахара. Для каждого из значений клеточный массив содержит отдельный слой, содержащий информацию только об одном из них, например: слой со значениями биомассы, слой со значением интенсивности запаха.

Препроцессор представляет из себя конвертер, задающий начальную конфигурацию клеточного массива. Поддерживаются режимы загрузки

клеточного массива из конфигурационного файла с особым синтаксисом, файла-сохранения, полученного после предыдущего запуска системы, и комбинированный, который позволяет загрузить часть клеточного массива из конфигурационного файла, часть — из файла-сохранения.

В постпроцессоре происходит обработка последнего состояния клеточного массива, сохранение состояния клеточного массива в файл (если задан соответствующий параметр), а также визуализация разных слоев клеточного массива, а именно — слоев с биомассой гриба и интенсивностью запаха, в файлы формата `pgm`.

В симуляторе непосредственно моделируется желаемое поведение. Было сформулировано несколько требований. Первое из них — моделирование поведения гриба, при котором он активно строит сеть, соединяющую разные участки, где находится еда (далее — сахар). Это поведение описывает такое свойство гриба, как нахождение кратчайших путей. Вторым требованием стало моделирование интенсивности запаха, исходящего от сахара. По задумке, запах должен быть «мотивацией» для активных действий гриба по распространению в направлении увеличения интенсивности запаха.

Основой для моделирования интенсивности запаха сахара стал механизм наивной диффузии, описанный в [3]. Моделирование происходит в три этапа.

1. Генерация: в клетке, где находится сахар, генерируется некоторое высокое значение интенсивности запаха.

2. Диффузия: выбирается случайный сосед, с которым происходит обмен некоторой частью значения интенсивности запаха (далее — часть обмена).

3. Исчезание: на каждой итерации значение интенсивности запаха сахара уменьшается на некоторое число процентов.

Такая модель позволила получить качественную картину распространения запаха. Стоит отметить, что если часть обмена достаточно мала (менее 0.2), то получаемая картина соответствует ожиданиям, иначе — начинает появляться автоматный шум, искажающий картину; однако стандартная процедура осреднения по значению в некотором радиусе нивелирует данный недостаток.

В процессе создания правил перехода для слоя биомассы было создано и протестировано множество моделей, однако каждая из них

обладала своими недостатками. В итоге был выбран вектор развития, при котором поведение гриба будет соответствовать заявленному требованию. В рамках этого направления было реализовано конкретное состояние гриба — существование в полностью изолированном пространстве. Данная модель была вдохновлена другой моделью — FNR-MP [4].

Моделирование поведения гриба описывается следующим образом: если количество биомассы гриба в текущей клетке выше некоторого критического значения, то данная клетка становится «растекающейся», выбирается сосед с минимальным количеством биомассы, и часть собственного значения передается ему. Иначе, если значение биомассы меньше критического, данная клетка становится «собирающейся»: выбирается сосед с наименьшим не равным нулю количеством биомассы, и часть собственного значения передается ему.

Так как моделирование поведения биомассы и интенсивности запаха сахара использует механизм обмена веществом, применяется асинхронный режим работы клеточного автомата. Есть возможность использовать синхронный режим с несколькими тактами, но это требует внесения изменений в текущую модель.

Был проведен эксперимент: в центр поля помещается большой фрагмент биомассы, который с течением времени равномерно разрастается в некоторое округлое пятно конечного радиуса, после чего рост останавливается и наблюдаются некоторые движения внутри пятна. Фронт увеличения пятна при росте гриба утолщен по сравнению с внутренней частью. Такое поведение соответствует реальному.

В дальнейшем планируется доработка модели до более общего вида, подбор правил перехода для формирования сил, способствующих перемещению биомассы, тестирование модели и сравнение с результатами известных натуральных экспериментов.

Литература

1. Горленко М. В., Бондарцева М. А., Гарибова Л. В. и др. Грибы СССР. Серия: Справочники-определители географа и путешественника. М.: Мысль, 1980. 303 с.
2. Medvedev Yu.G. Architecture of the Cellular-Automata Topologies Library // Bull. Nov. Comp. Center, Comp. Science. 2022. Iss. 46. P. 27–41.

3. Бандман О. Л. Клеточно-автоматные модели пространственной динамики // Системная информатика: Сб. науч. тр. Вып. 10. Новосибирск: Изд-во СО РАН, 2006. С. 59–113.

4. Медведев Ю. Г. Многочастичная клеточно-автоматная модель потока жидкости FHP-MP // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2009. № 1 (6). С. 33–40.

Научный руководитель — канд. техн. наук Ю. Г. Медведев,
ИВМиМГ СО РАН, НГУ, НГТУ

Реализация алгоритма планирования на вычислительных моделях с одномерными массивами с линейной индексацией

Р. С. Морозов

Новосибирский государственный университет

✉ r.morozov@g.nsu.ru

При решении задач численного моделирования можно столкнуться с проблемой низкой производительности компьютерных программ. Часто значительное ускорение может быть получено при использовании параллельных программ. Однако вручную сконструировать такую программу не всегда просто. Поэтому разрабатываются системы, позволяющие пользователю автоматически получить параллельную программу для решения своей задачи. Саму задачу в таком случае нужно сформулировать в понятном для системы виде. И один из подходов, предоставляющих такую возможность — использование вычислительных моделей [1].

Вычислительные модели позволяют частично описать предметную область в виде переменных и операций над ними. Эти объекты задают граф, по которому удобно определять зависимости по данным для автоматического построения параллельной программы. Задача на вычислительной модели представлена множествами заданных

и требуемых переменных. В общем случае в модели описано больше операций, чем необходимо для решения конкретной задачи. Поэтому для эффективного вычисления нужно выполнить планирование, то есть найти подграф, задающий решение задачи.

В работе рассматриваются вычислительные модели с одномерными массивами, индексы всех объектов связаны линейными зависимостями. Массовые операции позволяют задавать потенциально бесконечные множества объектов.

Описан алгоритм планирования [2] на таких вычислительных моделях, но на практике его реализация сталкивается с трудностями при представлении бесконечных множеств в памяти ЭВМ и применении к ним множественных операций.

В данной работе предлагается представление линейных множеств на основе арифметических прогрессий, которое учитывает их пересечения и хранит информацию в компактном виде по некоторым правилам. Единственное ограничение — объекты, не попадающие в бесконечные арифметические прогрессии, хранятся в виде отдельных элементов, что делает представление неэффективным при хранении длинных ограниченных последовательностей.

Также реализована часть алгоритма планирования, использующая предлагаемый класс для хранения данных при работе с массивами. Такая реализация позволяет выполнять планирование на некоторых простых примерах вычислительных моделей.

Литература

1. Вальковский В. А., Малышкин В. Э. Синтез параллельных программ и систем на вычислительных моделях. Новосибирск: Наука, 1988, 128 с.
2. Малышкин В. Э., Панкратов С. А. Алгоритм имитации планирования // Логические вопросы теории типов данных (Вычислительные системы). 1986. Вып. 114. С. 122–136.

Научный руководитель — канд. техн. наук, В. А. Перепёлкин,
ИВМиМГ СО РАН, НГУ, НГТУ

Разработка алгоритмов фильтрации данных FARICH-детектора

М. А. Муратов

Новосибирский государственный университет

✉ m.muratov@g.nsu.ru

Работа посвящена созданию алгоритмов для фильтрации данных FARICH-детектора [1]. В основе детектора лежит фотоматрица, которая улавливает фотоны черенковского излучения. Поток данных содержит информацию о срабатывании элементов фотоматрицы, при этом значительное количество срабатываний является шумовыми. Необходимо различать в потоке группы сигнальных срабатываний, обусловленных воздействием черенковского излучения. Разработан алгоритм фильтрации потока данных, основанный на применении статистического анализа. Используется предположение о том, что среднее время между сигнальными срабатываниями меньше среднего времени между срабатываниями во всем потоке данных. Кроме того, пространственное распределение элементов фотоматрицы, срабатывающих от группы событий, носит особый характер: образует эллиптическую форму на фотоматрице.

Для разных параметров шума были выведены параметры фильтрующего алгоритма, при которых, с одной стороны, сохранялось как можно больше сигнальных срабатываний, а с другой — было как можно меньше шумовых элементов данных.

В ходе проведения испытаний программы было установлено, что существенную долю времени работы программы занимают операции доступа в память.

Литература

1. Barnyakov A. Yu. et. al. FARICH detector beam test results // J. of Instrumentation. 2020. Vol. 15, iss. 10. P. C10014, DOI: 10.1088/1748-0221/15/10/C10014.

Научный руководитель — М. А. Городничев, ИВМиМГ СО РАН, НГУ, НГТУ

Разработка и реализация алгоритмов автоматической генерации LuNA-программ из последовательных программ на основе дополнительной информации о возможности их распараллеливания и фрагментации

В. К. Синюков

Новосибирский государственный университет

✉ v.sinyukov@g.nsu.ru

Задача автоматической генерации параллельных программ из последовательных — это в общем случае задача алгоритмически труднорешаемая. Поэтому, если такая задача и ставится, то чаще всего для какой-то конкретной предметной области. Работа направлена на решение данной задачи на примере системы фрагментированного программирования LuNA [1], которая разрабатывается в ИВМиМГ СО РАН. В качестве предметной области были выбраны методы численного моделирования, подобные методу Якоби решения систем линейных алгебраических уравнений. Здесь имеется в виду структурное подобие, то есть вышеописанные методы распараллеливаются аналогичным образом. Важно отметить, что целью данного проекта является не создание новой системы автоматического распараллеливания, а адаптация готовой системы, в качестве которой была выбрана система Sapfor [2], разрабатываемая в ИПМ им. Келдыша. Благодаря такой адаптации появится возможность автоматически генерировать LuNA-программы из последовательных программ, реализующих алгоритмы и методы численного моделирования. Это, во-первых, позволит получать LuNA-программы без необходимости изучения системы LuNA и параллельного программирования в целом. Во-вторых, это позволит пополнять фонд LuNA-программ с меньшими трудозатратами. И, в-третьих, может облегчить изучение системы начинающему LuNA-программисту, который сможет в случае затруднения поиска ошибок в своей написанной вручную LuNA-программе сгенерировать программу авто-

матически и путем сравнения двух программ понять, что он делал неправильно.

Была выбрана следующая схема автоматической генерации LuNA-программ из последовательных (см. рисунок): исходная программа (последовательная) на языке C, C++ или Fortran поступают в Sapfor и фрагментатор. Sapfor извлекает из последовательного кода информацию, необходимую для дальнейшего распараллеливания. Фрагментатор — это модуль, задачей которого является принятие ряда решений о способе фрагментирования на основе информации, собранной Sapfor, дополнение этой информации и ее фиксация в определенном формате (на схеме — «список знаний»). Далее остается только по исходной программе и списку знаний технически сгенерировать LuNA-программу, это делает генератор.

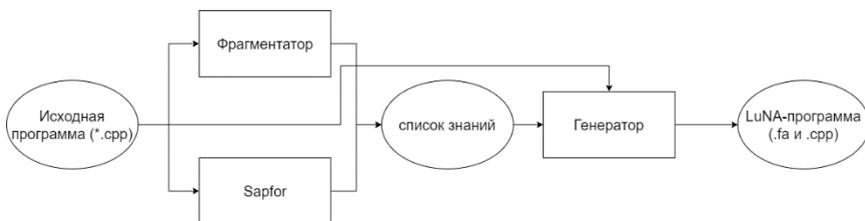


Схема автоматической генерации LuNA-программ из последовательных программ

Исходя из описания данного процесса выделены следующие задачи:

- определение модели данных передачи информации из Sapfor в LuNA;
- определение интерфейса передачи информации из фрагментатора в генератор;
- разработка и реализация фрагментатора;
- разработка и реализация генератора.

В рамках работы выполняется разработка и реализация генератора на основе результатов предыдущих работ по определению интерфейса передачи информации из фрагментатора в генератор, основанного на текстовом формате JSON (JavaScript Object Notation). Генератор реализован на языке программирования Java. В результате работы алгоритмы генерации были разработаны и реализованы для част-

ных случаев, то есть для некоторых последовательных программ уже имеется возможность генерации LuNA-программ по исходному листингу и JSON-списку знаний.

Направления дальнейшего развития работы:

- продолжение реализации генератора для других классов приложений;
- уточнение и исправление интерфейса передачи информации из фрагментатора в генератор по мере необходимости;
- определение модели данных передачи информации из Sapfor в LuNA и реализация фрагментатора.

Литература

1. Malyshkin V.E., Perepelkin V.A. LuNA Fragmented Programming System, Main Functions and Peculiarities of Run-Time Subsystem // Proceedings of the 11th International Conference on Parallel Computing Technologies, LNCS. 2011. Vol. 6873. P. 53–61.

2. Kataev N.A. Application of the LLVM Compiler Infrastructure to the Program Analysis in SAPFOR // Суперкомпьютерные дни в России: Труды международной конференции (24–25 сентября 2018 г., г. Москва), М.: Изд-во МГУ, 2018, P. 76–88.

Научный руководитель — канд. техн. наук В. А. Перепёлкин,
ИВМиМГ СО РАН, НГУ, НГТУ

Автоматическое создание графа зависимостей для статического анализатора LuNA-программ

В.Д. Царев

Новосибирский государственный университет

✉ v.tsarev@g.nsu.ru

В течение последних лет в ИВМиМГ СО РАН ведется разработка технологии фрагментированного программирования — технологии, позволяющей упростить создание эффективных параллельных про-

грамм. Ее реализация — язык программирования и система LuNA. В силу специфичности языка потенциальные ошибки LuNA-программиста могут быть трудны в поиске и устранении. Во многих языках программирования бороться с этой проблемой помогают инструменты статического анализа кода, поэтому, в силу отсутствия таких инструментов в системе LuNA, возникает необходимость создать аналогичные инструменты и для нее.

Статический анализ выполняется над промежуточным представлением программы. Таким представлением может служить абстрактное синтаксическое дерево или граф потока управления [1], однако специфика программы на языке LuNA делает удобным использование такого представления, как граф зависимостей по данным [2]. Для системы LuNA была выбрана вариация, представляющая собой ориентированный граф, вершинами которого служат операторы языка LuNA, а ребрами — зависимости операторов по фрагментам данных. Такой граф удобен, поскольку зависимости по фрагментам данных в LuNA-программе диктуют порядок выполнения операторов, что позволяет легко обнаруживать конфликты между операторами или ошибки в инициализации фрагментов данных. Статический анализатор должен построить граф зависимостей по данным из выданной ему программы и провести проверки на ошибки, после чего сообщить пользователю о всех найденных недочетах. Построение графа представляет собой обход абстрактного синтаксического дерева программы с порождением и последующим связыванием вершин, после чего несколько функций (по одной на каждый тип ошибки) обходят граф в поисках недочетов.

В результате работы была создана программа, способная сгенерировать граф зависимостей по данным исходной программы на языке LuNA, содержащей исключительно операторы применения фрагмента кода. В дальнейшем планируется реализация создания графа и для остальных операторов языка, а также автоматизация поиска различных ошибок с помощью созданного графа и, в частности, вывод их пользователю в удобном формате, что является необходимым элементом анализатора в силу несовершенства текущего компилятора LuNA.

Литература

1. Компиляторы: принципы, технологии и инструментарий / А. В. Ахо, М. С. Лам, Р. Сети, Дж. Д. Ульман. — 2-е изд., пер. с англ. — М. : ООО «И.Д. Вильямс», 2008. — 1184 с.

2. Kuck D. J., Kuhn R. H., Padua D. A., Leasure B., Wolfe M. Dependence graphs and compiler optimizations // Proceedings of the 8th ACM SIGPLAN-SIGACT symposium on Principles of Programming Languages (POPL), pp. 207-218, 1981.

Научный руководитель — канд. техн. наук А. Ю. Власенко,
ИВМиМГ СО РАН, НГУ, НГТУ

СОДЕРЖАНИЕ

Предисловие	3
-------------------	---

Программа лекций Всероссийской летней XLI молодежной школы-конференции по параллельному программированию

Перепёлкин В. А. Об автоматическом синтезе параллельных программ на основе баз активных знаний	5
Власенко А. Ю. Методы отладки параллельных программ.....	6
Медведев Ю. Г. Моделирование пространственной динамики клеточными автоматами	6
Ильин В. П. Параллельные методы декомпозиции областей с графовыми предобуславливателями для решения многомерных краевых задач.....	7

Программа лекций Летней XLIII школы-конференции по параллельному программированию

Перепёлкин В. А. База активных знаний как практичный способ представления множества программ предметной области.....	9
Медведев Ю. Г. Основы клеточно-автоматного моделирования природных процессов.....	9
Городничев М. А. Подходы к высокоуровневой организации расчетов на высокопроизводительных вычислительных системах.....	10
Пененко А. В. Современные методы моделирования химической кинетики в мультифизических задачах.....	11
Малышкин В. Э. От компиляции к генерации параллельных программ	11
Беляев Н. А. Особенности разработки и реализации средств автоматического конструирования параллельных программ	12

Тезисы докладов участников

Бурнышев Е. К. Разработка программной системы клеточно-автоматного моделирования газовых потоков.....	13
Глинский К. А. Профилирование библиотеки клеточно-автоматных топологий для синхронных клеточных автоматов на мультимпьютере	15
Голыжбина Ю. Ю. Разработка модулей детекции микросейсмических событий для суперкомпьютерного приложения «Микросейсмический мониторинг»	17
Ерошенко Д. С. Разработка системы моделирования организма «Физарум полицефалум».....	19
Морозов Р. С. Реализация алгоритма планирования на вычислительных моделях с одномерными массивами с линейной индексацией	22
Муратов М. А. Разработка алгоритмов фильтрации данных FARICH-детектора	24
Синюков В. К. Разработка и реализация алгоритмов автоматической генерации LuNA-программ из последовательных программ на основе дополнительной информации о возможности их распараллеливания и фрагментации.....	25
Царев В. Д. Автоматическое создание графа зависимостей для статического анализатора LuNA-программ.....	27

Научное издание

ВСЕРОССИЙСКАЯ ЛЕТНЯЯ ХLI МОЛОДЕЖНАЯ
ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 3–14 июля 2023 года)

•

ЛЕТНЯЯ ХLIII ШКОЛА-КОНФЕРЕНЦИЯ
ПО ПАРАЛЛЕЛЬНОМУ ПРОГРАММИРОВАНИЮ

(Новосибирск, 1–12 июля 2024 года)

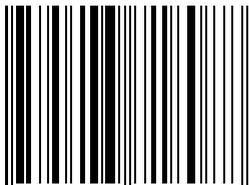
Тезисы докладов

Подготовка к печати *А. С. Терешкиной*
Верстка, обложка *А. С. Терешкиной*

Подписано к публикации 16.02.2026 г.
Формат 60 × 84 1/16. Уч.-изд. л. 2. Усл. печ. л. 1,8.
Заказ № 40.

Издательско-полиграфический центр НГУ
630090, Новосибирск, ул. Пирогова, 2.

ISBN 978-5-4437-1808-8



9 785443 718088