

Федеральное государственное учреждение  
«Федеральный исследовательский центр  
Институт прикладной математики им. М.В. Келдыша  
Российской академии наук»

*На правах рукописи*

---

*подпись соискателя*

**Тарасов Никита Игоревич**

**Разработка численных алгоритмов и параллельных программ для  
моделирования некоторых задач промышленной экологии**

Специальность 1.2.2 - математическое моделирование,  
численные методы и комплексы программ

Диссертация на соискания ученой степени  
кандидата физико-математических наук

Научный руководитель  
доктор физико-математических наук  
Поляков Сергей Владимирович

Москва – 2022

## Оглавление

<b>Введение .....</b>	<b>4</b>
<b>Глава 1. Математические модели и постановки задач .....</b>	<b>17</b>
1.1 Перечень моделей и расчетных блоков .....	17
1.1.1 Модели течения .....	19
1.1.2 Модель теплопроводности .....	24
1.1.3 Модель эволюции концентрации .....	25
1.1.4 Электростатическая модель .....	25
1.1.5 Микромасштабная модель .....	27
1.1.6 Многомасштабный подход, связь микро- и макромасштабов .....	29
1.1.7 Пересчет к размерным переменным .....	32
1.2 Дополняющие соотношения .....	35
1.2.1 Модель течения .....	35
1.2.2 Модель теплопроводности .....	36
1.2.3 Модель эволюции концентрации .....	37
1.2.4 Электростатическая модель .....	38
1.3 Постановки задач .....	38
1.3.1 Задачи верификации .....	39
1.3.2 Задача валидации .....	41
1.3.3 Модельные задачи .....	41
1.4 Выводы по Главе 1 .....	44
<b>Глава 2. Численные методы и их реализация .....</b>	<b>46</b>
2.1 Численные методы .....	46
2.1.1 Численные методы макроскопического уровня .....	46
2.1.2 Численные методы микроскопического уровня .....	50
2.1.3 Переход от макроскопического уровня к микроскопическому и обратно .....	51
2.2 Программная реализация .....	52
2.2.1 Параллельные алгоритмы решения задач .....	53
2.2.2 Расчетные коды .....	55

2.3 Выводы по Главе 2 .....	69
<b>Глава 3. Цифровая платформа.....</b>	<b>70</b>
3.1 Общие положения и предпосылки .....	70
3.2 Архитектура цифровой платформы .....	72
3.3 Прототип цифровой платформы.....	76
3.3.1 Технологический стек .....	77
3.3.2 Структура базы данных .....	79
3.3.3 Веб-сервер .....	81
3.3.4 Пользовательский веб-интерфейс.....	82
3.5 Выводы по Главе 3 .....	101
<b>Глава 4. Вычислительные эксперименты .....</b>	<b>102</b>
4.1 Задачи верификации .....	102
4.1.1 Установление течения Пуазейля.....	102
4.1.2 Перенос концентрации.....	104
4.1.3 Тепловое конвективное движение .....	106
4.2 Задача валидации .....	107
4.3 Модельные задачи.....	110
4.3.1 Задача об электромагнитной очистке водной среды .....	110
4.3.2 Задача о сорбционной очистке воздушной среды.....	112
4.3.3 Задача о загрязнении тепловыделяющего элемента .....	114
4.6 Выводы по Главе 4.....	119
<b>Заключение.....</b>	<b>121</b>
<b>Список литературы .....</b>	<b>125</b>

## Введение

Термин «экология» впервые введен Э. Геккелем в 1866 году [1] в смысле науки о взаимодействии живой и неживой природы – растений и животных. Развитие данного термина, обусловленное технологическим, социальным и научным прогрессом, привело к образованию отдельных областей, распределенных по существенно расширившимся объектам исследования. На сегодняшний день принято выделять такие разделы, как экология человека, экология растений, экология микроорганизмов и тому подобные, одним из которых является промышленная экология, включаемая в образовательную программу ВУЗ по таким различным направлениям, как, например, «Химическая технология» или «Оптотехника» [2 – 4].

Предметом промышленной экологии является воздействие промышленности на окружающую среду, в том числе последствия данного воздействия на качество жизни человека и общества в целом, и наоборот – воздействие внешней среды на технологические или сельскохозяйственные предприятия [5]. Основными целями, при этом, являются:

- выработка подходов рационального, ответственного природопользования;
- разработка принципов безотходных производств, в том числе технологий замкнутого цикла;
- предотвращение техногенных катастроф, а также путей восстановления окружающей среды.

Для выполнения поставленных целей в рамках промышленной экологии рассматриваются следующие задачи:

- стандартизация эксплуатации промышленных предприятий и выработка государственных стандартов [6], включающих нормы по допустимым загрязнениям воздуха, воды и почвы различными видами воздействий (химическому, вибрационному, электромагнитному, энергетическому);

- исследование последствий загрязнения, его влияния на флору и фауну;
- прогнозирование последствий деятельности промышленных предприятий, их влияния на окружающую среду;
- разработка систем очистки воздуха, воды и почвы, как для контроля выбросов во внешнюю среду, так и для использования в производственном процессе;
- вопросы эколого-экономической экспертизы, утилизации отходов и др.

Рассмотрим подробнее задачу очистки окружающих сред, решение которой с одной стороны позволяет соблюдать установленные нормы по допустимым выбросам, при сохранении эффективности хозяйственной деятельности, с другой стороны, сгладить воздействие окружающей среды на промышленность. Одним из способов решения данной проблемы, при рассмотрении водной и воздушной сред, является применение фильтрационных сооружений, состоящих из многоуровневых систем очистки. При этом используются различные методики и физические процессы, включающие фильтры, основанные на механических, тепловых, химических и электромагнитных эффектах, а также смешанный, гибридный подход их объединяющий. Если рассмотреть научную часть проблемы очистки сред от загрязнений, то на настоящем этапе исследований речь идет о комплексном подходе к проблеме, а именно, о создании технологий замкнутого цикла и повышении их эффективности. В этом контексте при анализе технологий очистки воды можно указать работы [7 – 15], воздуха – [16 – 20], а также многие другие.

Данная работа посвящена развитию многомасштабного численного подхода к моделированию этапа «тонкой» фильтрации среды, следующего за прохождением механических и/или химических фильтров. На данном этапе очистки водная среда подвергается электромагнитному или термическому воздействию, а воздушная среда проходит через гранулированные активные

вещества. Этап тонкой фильтрации актуален не только для выполнения норм экологической безопасности по стокам и выбросам промышленных установок и объектов. Он является зачастую необходимой частью самого производства. Например, в атомной энергетике необходимо получение теплового агента с предсказуемыми физико-химическими параметрами. В авиационной и автомобильной промышленности требуется специальная очистка воды, а также жидких и газообразных топлив, для увеличения срока службы и соблюдения стандартов качества при производстве. В фармакологии требуется особо чистая вода для производства лекарств. Отдельная потребность в тонкой очистке связана с разработкой средств индивидуальной защиты.

**Очистка водной среды.** С учетом того, что вода является хорошим растворителем, вопрос тонкой фильтрации включает в себя не только удаление из среды загрязняющей взвеси, но и растворенных в ней веществ (например, оксидов и солей металлов). Отметим также, что в засушливых регионах актуально опреснение водной среды от излишней солености, в том числе для получения питьевой воды. При этом используются следующие физико-химические процессы:

- мембранная фильтрация для осмотической очистки;
- сорбционная фильтрация с использованием пористых материалов;
- электромагнитная фильтрация, основанная на электролизе или омагничивании жидкости.

Мембранная очистка, применяемая при опреснении или очистке водной среды, основана на диффузионных процессах, возникающих при прохождении потока через тонкую пленку, пропускающую раствор с пониженным содержанием растворенного вещества и задерживающую частицы загрязнителя. При этом основным параметром варьирования пропускной способности очистного сооружения является изменение его размера, что не всегда является конструкционно возможным и экономически

оправданным.

При сорбционной фильтрации в качестве материала изделия используются адсорбционные (концентрирующие загрязнитель на поверхности) или абсорбционные (поглощающие загрязнитель в объеме) материалы, включенные в пористую основу, что позволяет добиться оптимального взаимодействия раствора с активным веществом. При этом очистка и повторное использование после предельного загрязнения данного типа фильтрующего устройства не представляется возможной, поскольку это может привести к повреждению пористого материала или утере содержащихся в нем активных элементов.

При применении обоих вышеуказанных способов фильтрации появляются дополнительные высокотоксичные отходы, утилизация которых является отдельной не менее сложной научной и практической проблемой.

Идея электромагнитной фильтрации заключается в разделении водного потока на нейтральные, положительно и отрицательно заряженные компоненты. При воздействии на поток магнитного и электрического полей возникает сила Лоренца, которая способствует такому разделению. Технически воздействие магнитного поля реализуется путем применения сильных постоянных магнитов или соленоидов, окружающих трубопровод. Электрическое воздействие реализуется путем погружения в трубопровод необходимого числа электродов.

Цель электромагнитной фильтрации – локализация заряженных частиц или растворенных ионов загрязнителей в предсказуемой области, откуда обогащенный ими раствор возможно либо отвести, либо задержать на продолжительное время, получив на выходе из системы очищенную воду. При этом модуляция качества очистки возможна посредством регулирования силы электромагнитного воздействия. Дополнительным преимуществом данного способа очистки является возможность эффективной регенерации очистной системы, а также извлечение из примесей полезных веществ

пригодных для вторичной переработки (например, извлечение из морской воды редкоземельных и благородных металлов).

**Очистка воздушной среды.** При очистке воздушной среды основной целью тонкой фильтрации является удаление из потока наноразмерных частиц загрязнителя. Удаление таких частиц на ранних стадиях очистки физически невозможно ввиду их особо малого размера. При этом опасность присутствия таких частиц в воздушном потоке связана с тем, что при попадании в легкие человека и животных, они остаются в биологических тканях на продолжительное время и приводят в итоге к развитию необратимых патологий органов дыхания.

Для очистки воздушного потока от твердых наночастиц могут применяться мембранные фильтры с дискретизацией малого размера. Подобные системы мембран представляют собой достаточно хрупкие устройства, подверженные загрязнению без возможности восстановления пропускной способности и эффективности фильтра.

Существуют альтернативные системы фильтрации газов с использованием гранулированных сорбентов – элементов различной формы, имеющих поверхностный или объемный заряд, позволяющий захватывать загрязняющие частицы малого размера. Данный подход позволяет управлять допустимой концентрацией загрязнителя и режимом работы фильтра с помощью увеличения количества сорбента. Кроме того, подобные очистные системы допускают эффективную регенерацию без опасности нарушения последующей работы.

**Технологии замкнутого цикла.** Идея, стоящая за разработками технологий замкнутого цикла, заключается в уменьшении интенсивности генерации отходов, а также в их использовании в целях производства. Такой подход не только увеличивает экономическую эффективность той или иной производственной цепочки, но и снижает экологическую нагрузку на окружающую среду [21].

Так, например, железистые загрязнители водной среды при накоплении

в фильтрующем устройстве допускают восстановление некоторого количества материала. Аналогичные процессы допустимы для основных загрязнителей воздуха – карбидной взвеси.

Кроме того, для соответствия стандартам безотходного производства сами фильтрационные установки должны быть не только ремонтпригодны, но и допускать восстановление и повторное использование функциональных элементов.

**Актуальность темы исследования** обусловлена необходимостью наличия инструмента моделирования процессов, проходящих в прецизионных (обеспечивающих высокую степень очистки от мелкодисперсных примесей, в том числе микро- и наноразмерных) фильтрационных установках, как с целью разработки новых высокоэффективных конструкций, так и для обслуживания уже существующих очистных систем. Во втором случае создание цифровых двойников действующих очистных установок позволит прогнозировать их состояние и выработать оптимальный график технического обслуживания, в том числе периодичность очистки или замены фильтров.

С математической точки зрения комплексное компьютерное моделирование процессов очистки находится в процессе становления. В частности, остается актуальной проблема расчета гидро- и газодинамических процессов в пористых и сыпучих средах реальной трехмерной геометрии. Также является актуальным учет в макроскопических моделях физико-химических процессов, происходящих на микроуровне. Проблема в целом решается применением различных многомасштабных подходов, однако их единичные реализации и разнородность математических описаний не позволяют пока говорить о надежности получаемых численных результатов. Отчасти это объясняется обилием постановок фундаментальных и прикладных задач, инициируемых научными группами и промышленностью, а также большим объемом предварительных исследований необходимых для

принятия тех или иных вычислительных моделей и компьютерных технологий.

**Степень научной разработанности**, поставленной в рамках настоящей диссертации задачи, с одной стороны заключается в хорошей обоснованности применяемых математических моделей и численных алгоритмов, обсуждаемых далее. С другой стороны, в рассматриваемой предметной области отсутствуют аналогичные программные пакеты комплексного многомасштабного моделирования процессов фильтрации. Зачастую применяются методы схематичного описания установок, не учитывающие геометрических особенностей фильтрационных компонент, в совокупности с упрощенными математическими моделями, во многом основанными на эмпирических и эвристических данных [22, 23].

**Целью данной работы** является разработка математических основ, программного вычислительного ядра и цифровой платформы для компьютерного и суперкомпьютерного моделирования процессов прецизионной фильтрации водной и воздушной сред от загрязняющих нейтральных и заряженных мелкодисперсных примесей.

**Задачами исследования** являются:

- анализ существующих промышленных методов и систем очистки водной и воздушной сред от загрязняющих примесей, выбор актуальных проблем в классе технологий, обеспечивающих прецизионное качество, формулировка модельных задач экологии;
- обзор методов математического моделирования фильтрационных процессов для выбранных постановок задач, разработка комплексного многомасштабного подхода, формулировка конкретных математических моделей на основе уравнений механики сплошной среды и дополнение их материальными уравнениями среды, сформированными на основе методов частиц и молекулярной динамики;
- разработка численных методов и параллельных алгоритмов решения выбранных двух- и трехмерных модельных задач в рамках

макроскопических описаний с акцентом на использование неструктурированных гибридных сеток, а также численных процедур уточнения материальных уравнений среды и параметров уравнений методами частиц;

– программная реализация численных методов решения базовых уравнений механики сплошной среды в двухмерной и трехмерной постановке, разработка вычислительных кодов для решения выбранных модельных задач;

– разработка архитектуры и программная реализация цифровой платформы, обеспечивающей полный цикл компьютерного моделирования;

– проведение серии вычислительных экспериментов с целью верификации и валидации разработанных моделей, численных методов, алгоритмов и программ.

**Научной новизной** диссертационной работы являются разработанные комплексные математические модели, предназначенные для моделирования процессов тонкой очистки водной и воздушной сред, их численные алгоритмы и программная реализация, а также цифровая платформа для проведения вычислительных экспериментов на вычислительных кластерах и суперкомпьютерах.

**Теоретическая значимость** работы состоит в создании комплексных математических макроскопических моделей экологической направленности, параметры которых можно уточнить в рамках использования многомасштабного подхода, объединяющего классические модели и методы механики сплошных сред и динамики частиц.

**Практическая ценность исследования** заключается в разработанном прототипе цифровой платформы и комплексе параллельных программ для суперкомпьютерного моделирования процессов очистки водной и воздушной сред от загрязняющих их примесей методами макроскопического уровня или в рамках многомасштабных вычислений.

**Методология и методы исследования.** В части математического моделирования разработанная методология заключается в использовании квазигидро- и квазигазодинамического систем уравнений для описания многокомпонентных двухфазных течений жидкости и газа (в качестве второй фазы включающих твердые мелкодисперсные микро- и наночастицы) в областях сложной геометрии близкой к реальной. При этом твердая фаза описывается на макроскопическом уровне в терминах концентраций примесей и использует системы уравнений конвекции-диффузии-реакции, дополненные электростатическими, тепловыми и реакционными членами.

В качестве основы комплексного численного алгоритма применяется метод конечного объема по пространственным переменным. При этом контрольные объемы совпадают с используемыми сеточными элементами. Для дискретизации по времени используются явные схемы. Многомасштабность заключается в моделировании процессов на макроскопическом уровне на основе соответствующих систем уравнений, параметры которых (кинетические коэффициенты уравнений, потоки на границах, коэффициенты сорбции и теплопередачи) и константы материальных уравнений (в том числе уравнений состояния), либо определены на основе методов частиц и молекулярной динамики заранее, либо рассчитываются непосредственно в рамках многомасштабного подхода на этапе так называемых подсеточных вычислений. Наиболее важным элементом предлагаемого многомасштабного подхода является расчет процессов сорбции и теплообмена на твердых поверхностях элементов систем очистки (стенок резервуаров и каналов, внешней и внутренней поверхностей гранул сорбентов).

При программной реализации разработанных комплексных моделей сформировано необходимое вычислительное ядро, которое представляет собой набор классов и методов, основанных на объектно-ориентированном и шаблонном подходах. Цифровая платформа для проведения вычислительных экспериментов реализуется посредством применения веб-технологий в виде

клиент-серверного решения, использующего в качестве клиентской части браузер на компьютере пользователя, а в качестве сервера – среду из одного или нескольких веб-серверов, размещенных в сети Интернет.

**Достоверность и обоснованность** диссертации обусловлена использованием метода расщепления по физическим процессам для формирования комплексных моделей процессов фильтрации. При этом элементы их составляющие представляют собой корректные постановки начально-краевых задач, основанные на хорошо известных, обоснованных и неоднократно проверенных математических подходах, в том числе классических моделях сплошной среды и моделях динамики частиц. Немаловажным также является применение консервативного сеточного метода конечного объема при конструировании итогового вычислительного алгоритма.

#### **Положения, выносимые на защиту**

1) Разработаны комплексные математические модели, применяемые в задачах экологической направленности (процессы фильтрации, загрязнение функциональных элементов конструкции), а именно:

- а) квазигидродинамическая модель течения водной среды с учетом динамики электромагнитного поля и процессов конвекции и диффузии заряженных примесей;
- б) квазигазодинамическая многокомпонентная модель течения воздушной среды с учетом наличия в ней твердых наноразмерных частиц;
- в) квазигидродинамическая многокомпонентная модель течения жидкой среды с учетом процессов конвекции, диффузии и химических превращений,

и способы их уточнения и агрегирования в рамках многомасштабного подхода;

2) предложены численные методы анализа разработанных комплексных моделей, основанные на методе конечных объемов на

неструктурированных сетках, и их программные реализации, ориентированные на использование современных компьютерных и суперкомпьютерных вычислительных систем;

3) создана цифровая платформа, включающая вычислительное ядро, базирующееся на разработанных комплексных моделях, и дополнительный программный инструментарий, реализующий всю цепочку вычислительного эксперимента при решении актуальных прикладных задач экологической направленности;

4) проведено численное моделирование процессов фильтрации для ряда практических задач промышленной экологии, результаты которого имеют самостоятельное значение, а также подтверждают эффективность разработанной компьютерной технологии.

**Апробация результатов** работы была проведена посредством личного представления автором результатов на ряде научных конференций:

- Международная конференция "Mathematical Modeling and Computational Physics, 2019" (ММСР2019) (Татры, Словакия, 1-5 июля 2019),
- IV Всероссийская научная конференция "Теплофизика и физическая гидродинамика" (Ялта, Россия, 15-22 сентября 2019 г.),
- 7th European Seminar on Computing (ESCO 2020) (Пльзень, Чехия, 8-12 июня 2020 г., удаленно),
- XIII международная конференция «Сеточные методы для краевых задач и приложения» (Казань, Россия, 20-28 октября 2020 г., удаленно),
- Fourth Virtual Workshop on Numerical Modeling in MHD and Plasma Physics: Methods, Tools, and Outcomes (Новосибирск, 12- 14 октября 2021 г., удаленно).

В рамках данной работы было **опубликовано** 13 статей [24 – 36] в том числе 10 из списка ВАК, 9 индексируются в базах Web of Science и Scopus.

**Личный вклад** состоит формировании набора комплексных математических моделей для исследования процессов тонкой фильтрации, разработке общего численного алгоритма, основанного на классическом

методе конечного объема, программной реализации вычислительного ядра и цифровой платформы.

Работа состоит из введения, четырех глав, заключения, списка литературы и приложения. Диссертация содержит 139 страниц и 71 рисунок.

**В первой главе** рассматриваются ключевые физико-химические процессы характерные для задач промышленной экологии. На основании этого конструируется общий многомасштабный подход моделирования фильтрационных установок, происходит выбор моделей, определяются масштабные соотношения. Устанавливается набор верификационных модельных задач.

**Вторая глава** посвящена разработке численных методов для выбранных моделей, а также приведено описание их программной реализации, формирующей вычислительное ядро.

**В третьей главе** обсуждается построение цифровой платформы, определяется набор технологий и библиотек, рассматриваются особенности ее реализации. В данной главе также представлены основные сценарии взаимодействия с прототип цифровой платформы для проведения полного цикла компьютерного моделирования.

**Четвертая глава** рассматривает вопросы верификации и валидации. Приводятся результаты расчетов модельных задач, определенных в первой главе.

**В заключении** подробно рассматриваются основные результаты диссертационной работы, пути дальнейшего развития вычислительного ядра и цифровой платформы в целом.

Автор выражает особую благодарность научному руководителю д.ф.-м.н. Полякову Сергею Владимировичу за неоценимую помощь и мотивацию при подготовке данной диссертации.

Автор глубоко благодарит соавторов д.ф.-м.н. Карамзина Ю.Н., к.ф.-м.н. Кудряшову Т.А., д.ф.-м.н. Подрыгу В.О. и Пузырькова Д.В. за совместную продуктивную работу над текстами публикаций.

Автор признателен д.ф.-м.н. Гасилову В.А., к.ф.-м.н. Ольховской О.Г., Бойкову Д.С. и к.ф.-м.н. Ускову Р.В. за ценные обсуждения подходов проектирования прикладных вычислительных комплексов и совместную выработку требований, накладываемых на цифровые платформы.

Автор выражает глубокую признательность Храмцову М.Ю. за ценные советы по принципам работы с операционными системами UNIX и системой управления прохождением пользовательских задач – СУППЗ.

## **Глава 1. Математические модели и постановки задач**

### **1.1 Перечень моделей и расчетных блоков**

При прохождении среды через фильтр происходит множество физических, электро- и термохимических эффектов, обеспечивающих ее очистку. Для адекватного описания данного процесса необходимо подробнее рассмотреть принципы работы устройств тонкой фильтрации, что позволит определить список основных моделей для дальнейшего получения численного алгоритма и его программной реализации.

Рассмотрим сначала общий процесс прохождения водной или воздушной среды через фильтрационное устройство. При попадании загрязненного вещества в исследуемый элемент конструкции она проходит через активную область фильтра (подверженную электромагнитному, термическому воздействию), основной задачей которого является изменение однородного распределения загрязнителя на стратифицированное. Полученная в активной области неоднородность позволяет сформировать два подхода для очистки среды:

- применение в областях локализации примеси сорбционных элементов, допускающих накопления загрязнителя в течении некоторого времени эффективной эксплуатации, в пределах которого происходит забор очищенной среды;

- геометрическая сепарация потока и оснащение фильтрационного устройства отдельными выходными отверстиями, при котором насыщенная примесью среда покидает область исследования в одной части, а забор очищенного вещества осуществляется в другой.

Исходя из рассмотренных фактов для построения полноценной модели фильтрации, необходимо учитывать следующие процессы:

- течение среды в исследуемой области, в том числе имеющей сложную геометрию;

- конвекционно-диффузионный перенос загрязнителя;
- учет внешних эффектов, приводящий к неоднородности распределения загрязнителя (силы Лоренца, термохимические процессы);
- учет сорбции на стенках фильтра, специальных сорбционных элементах (гранулированных, пористых).

Комплексная модель, описывающая перечисленные эффекты, позволит с одной стороны проводить численные эксперименты с целью оптимизации геометрии и конструкционных материалов фильтра, а также режимов его работы, с другой – формировать цифровой двойник уже действующей установки, посредством которого возможно устанавливать периоды эффективной эксплуатации и технического обслуживания очистного устройства.

Подробнее рассмотрим перечисленные физические процессы – можно видеть, что существует большой разброс в их масштабах: размеры установок (1-100 м), размеры фильтрующих систем (1-100 см), размеры элементов фильтров и гранул (0.1-100 мм), размеры пор и каналов между ними (0.01-1 мм), размеры микро- и наночастиц загрязнителей (0.001-10 мкм). В связи с этим базовым подходом к моделированию течений среды и конвекционно-диффузионного переноса загрязнителей является механика сплошных сред (МСС) при котором исследуемое пространство воспринимается как непрерывный континуум. Под макроскопическим уровнем описания эволюции состояния этого континуума будем понимать те или иные модели МСС.

Разного рода процессы в тонких капиллярах и порах (например, процессы захвата частиц загрязнителя стенками капилляра или гранулы), а также взаимодействия отдельных атомов и молекул происходят на микроскопических масштабах вплоть до атомных. Этот микроскопический уровень описания может быть представлен статистическими моделями на основе метода Монте-Карло, кинетическими моделями Больцмана, моделями частиц, в том числе, моделями молекулярной динамики. В данной работе

были выбраны модели и методы частиц.

При объединении макроскопических и микроскопических описаний существуют два подхода. Первый из них предполагает использование результатов микроскопического моделирования в моделях макроскопического уровня. Например, неизвестные параметры веществ и материалов, уравнений состояния среды, кинетические коэффициенты уравнений гидродинамики и тепло- и массопереноса могут быть вычислены предварительно методами микроскопического уровня. Данный подход назовем упрощенным многомасштабным подходом.

Более точный, но и более вычислительно емкий подход предполагает совместное рассмотрение моделей макроскопического и микроскопического уровня. Его в литературе называют многомасштабным.

### **1.1.1 Модели течения**

#### **1.1.1.1 Течение водной среды**

Поведение водной среды, которая представляется вязкой несжимаемой жидкостью, в рамках модели сплошных сред описывается уравнениями Навье-Стокса с учетом условия несжимаемости [37]. Более того, известен факт того, что при небольших перепадах давления и малых скоростях (что справедливо при моделировании фильтрационных устройств) допускается описание поведения потока воздуха как несжимаемой среды. Это позволяет использовать единый подход при моделировании течения как водной, так и воздушной сред.

Численное решение уравнений Навье-Стокса с учетом условия несжимаемости связано с проблемой неустойчивости численных схем второго пространственного порядка точности при использовании центральных разностей [38]. Это приводит либо к необходимости формирования специальных численных подходов с разнесением физических величин (методики типа MAC или SIMPLE), либо к получению иной постановки исходящей из физико-математических следствий исходной

системы. Одной из возможных альтернативных моделей гидродинамики является применение метода двойного потенциала, заключающегося в переходе от естественных переменных (давление-плотность) к переменным векторный потенциал – скалярный потенциал – завихренность [39 – 41], что позволяет получить устойчивую численную схему, а кроме того, введением скалярного потенциала, избежать сложности постановки граничных условий на потенциал векторный. При использовании данной методики в ранних работах все же встретились определенные трудности при постановке граничных условий в областях сложной геометрии, а также областях, имеющих несколько выходных отверстий. Отметим также, что метод двойного потенциала для моделирования вязкого несжимаемого течения все еще слабо отражен в научной литературе для целей моделирования внутренних течений и применяется, зачастую, для получения конвективного движения в замкнутой области или рассмотрении задач обтекания.

Существует и иной подход для решения поставленной задачи, основанный на квазигидродинамической (КГиД) модели [42 – 45], в отличие от метода двойного потенциала нашедший широкое применение при исследовании широкого класса задач. Более того, использование КГиД модели обеспечивает возможность устойчивого счета даже на очень подробных сетках. Приведем далее систему уравнений для описания изотермического течения:

$$\frac{\partial \mathbf{u}}{\partial t} = \nabla \left[ \frac{1}{\text{Re}} (\nabla \otimes \mathbf{u} + (\nabla \otimes \mathbf{u})^T) + \mathbf{w} \otimes \mathbf{u} + \mathbf{u} \otimes \mathbf{w} - \mathbf{u} \otimes \mathbf{u} - p \right], \quad (1.1)$$

$$\nabla(\nabla p) = \nabla \left[ \frac{1}{\tau} \mathbf{u} - (\mathbf{u}, \nabla) \mathbf{u} \right], \quad (1.2)$$

$$\mathbf{w} = \tau [(\mathbf{u}, \nabla) \mathbf{u} + \nabla p], \quad (1.3)$$

где  $\frac{\partial}{\partial t}$  – временная производная,  $\text{Re} = \frac{\rho u_0 L_h}{\eta}$  – число Рейнольдса,

$\nabla = \left\{ \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right\}^T$  – оператор Гамильтона,  $\mathbf{u} = \{u_x, u_y, u_z\}^T$  – вектор

скорости,  $(\bullet \otimes \bullet)$  – прямое произведение векторов,  $p$  – давление,  $\mathbf{w}$  –

регуляризирующая поправка,  $\tau \sim \frac{1}{\text{Re}}$  – параметр регуляризации,  $\rho$  –

плотность,  $\eta$  – коэффициент динамической вязкости,  $u_0$  – характерная скорость течения,  $D_h$  – характерный линейный размер области (гидродинамический диаметр).

Система уравнений (1.1) – (1.3) дополняется граничными и начальными условиями, которые приведем далее.

### 1.1.1.2 Течение воздушной среды

В случае существенных отклонений параметров среды от несжимаемой жидкости в качестве модели предлагается использовать полную систему квазигазодинамических уравнений (КГД систему), записанную для смеси компонент [42 – 47], исключая твердую фазу. Для последней ниже используется уравнение конвекции-диффузии, которое при необходимости дополняется уравнением теплопроводности. В случае смеси газов система КГД уравнений записывается для каждого газа отдельно и имеет одинаковый вид [29]. Модель течения газа реализована вне рамок данной работы на базе программного комплекса GIMM\_NANO с использованием программных модулей GIMM\_APP\_QGD\_CPU [48] и GIMM\_APP\_QGD\_MD\_CPU [49].

Запишем КГД уравнения в трехмерном случае для смеси газов в инвариантном относительно системы координат виде в размерных переменных (ниже используется система единиц СИ) вместе с уравнениями связей и состояния:

$$\frac{\partial \rho_l}{\partial t} + \text{div } \mathbf{W}_l^{(\rho)} = 0, \quad (1.4)$$

$$\frac{\partial(\rho_l u_{l,k})}{\partial t} + \operatorname{div} \mathbf{W}_{l,k}^{(\rho u)} = S_{l,k}^{(\rho u)}, \quad k = x, y, z, \quad (1.5)$$

$$\frac{\partial E_l}{\partial t} + \operatorname{div} \mathbf{W}_l^{(E)} = S_l^{(E)}, \quad (1.6)$$

$$\left\{ \begin{array}{l} E_l = \frac{1}{2} \rho_l |\mathbf{u}_l|^2 + \rho_l \varepsilon_l, \quad \varepsilon_l = c_{V,l} T_l, \quad H_l = \frac{E_l + p_l}{\rho_l}, \quad p_l = Z_l \rho_l \mathfrak{R}_l T_l, \\ \gamma_l = \frac{c_{p,l}}{c_{V,l}}, \quad \operatorname{Pr}_l = \frac{\mu_l c_{p,l}}{\chi_l}, \quad \operatorname{Sc}_l = \frac{\mu_l}{\rho_l D_l}, \quad \operatorname{Ma}_l = \frac{|\mathbf{u}_l|}{a_l}, \quad \operatorname{Re}_l = \frac{\rho_l |\mathbf{u}_l| \lambda_l}{\mu_l}, \end{array} \right. \quad (1.7)$$

где все переменные с индексом  $l$  относятся к газу типа  $l$ , каждая компонента имеет свои числовую плотность (концентрацию)  $n_l$ , массовую плотность  $\rho_l = m_l n_l$  ( $m_l$  – масса молекул газа  $l$ ). Каждый газ также характеризуется своей температурой  $T_l$  и макроскопической скоростью  $\mathbf{u}_l$ . Другие параметры компонент смеси:  $p_l$  – парциальные давления газов в смеси;  $E_l$ ,  $H_l$  и  $\varepsilon_l$  – плотности полной энергии, энтальпии и внутренние энергии компонент смеси;  $\mu_l = \mu_l(T_l)$ ,  $D_l = D_l(T_l)$ ,  $\chi_l = \chi_l(T_l)$  – кинетические коэффициенты компонент смеси, а именно: коэффициенты динамической вязкости, диффузии и теплопроводности. Переменные  $Z_l = Z_l(T_l, \rho_l)$ ,  $\gamma_l = \gamma_l(T_l, \rho_l)$ ,  $c_{V,l} = c_{V,l}(T_l)$ ,  $c_{p,l} = c_{p,l}(T_l)$  и  $\mathfrak{R}_l = k_B / m_l$  – коэффициенты сжимаемости, показатели адиабат, удельные теплоемкости и индивидуальные газовые постоянные компонент смеси ( $k_B$  – постоянная Больцмана),  $\operatorname{Pr}_l$ ,  $\operatorname{Sc}_l$ ,  $\operatorname{Ma}_l$  и  $\operatorname{Re}_l$  – числа Прандтля, Шмидта, Маха и Рейнольдса для компонент смеси;  $\lambda_l$  – средние длины свободного пробега; векторы  $\mathbf{W}_l^{(\rho)}$ ,  $\mathbf{W}_{l,k}^{(\rho u)}$ ,  $\mathbf{W}_l^{(E)}$  с точностью до знака совпадают с потоками массовой плотности, соответствующих компонент плотности импульса и плотности энергии. Обменные члены  $S_{l,k}^{(\rho u)}$  и  $S_l^{(E)}$  учитывают перераспределения импульса и энергии между компонентами смеси.

Векторы потоков и обменные члены выглядят следующим образом:

$$\mathbf{W}_l^{(\rho)} = \rho_l \mathbf{u}_l - \mathbf{w}_l, \quad \mathbf{w}_l = \tau [\operatorname{div}(\rho_l \mathbf{u}_l \otimes \mathbf{u}_l) + \nabla p_l], \quad (1.8)$$

$$\begin{aligned} \mathbf{W}_{l,k}^{(\rho u)} &= \rho_l \mathbf{u}_l u_{l,k} + \mathbf{e}_k \left( p_l + \frac{2}{3} \mu_l \operatorname{div} \mathbf{u}_l \right) - \mu_l (\nabla u_{l,k} + (\nabla, \mathbf{e}_k) \mathbf{u}_l) - \\ &\quad - (\rho_l w_{l,k} \mathbf{u}_l + \rho_l \mathbf{w}_l u_{l,k}), \\ S_{l,k}^{(\rho u)} &= \nu_{ll'} \rho_l (u_{l,k}^* - u_{l,k}), \quad k = x, y, z; \end{aligned} \quad (1.9)$$

$$\begin{aligned} \mathbf{W}_l^{(E)} &= (\rho_l \mathbf{u}_l - \rho_l \mathbf{w}_l) H_l - \chi_l \nabla T_l + \left( \frac{2}{3} \mu_l \operatorname{div} \mathbf{u}_l \right) \mathbf{u}_l - \\ &\quad - \sum_{k=x,y,z} \mu_l (\nabla u_{l,k} + (\nabla, \mathbf{e}_k) \mathbf{u}_l) + (\rho_l \mathbf{w}_l, \mathbf{u}_l) \mathbf{u}_l, \\ S_l^{(E)} &= \nu_{ll'} \rho_l (E_l^* - E_l). \end{aligned} \quad (1.10)$$

Обменные члены содержат в качестве параметров компоненты скоростей  $\mathbf{u}_l^*$  и плотности энергии  $E_l^*$ . Эти параметры соответствуют равновесному состоянию смеси, но не реализуются при переходном процессе. Фактически они служат ориентиром для динамических переменных. Также в обменных членах присутствуют  $\nu_{ll'}$  – частоты взаимных столкновений между молекулами сорта  $l$  с молекулами сорта  $l'$ , которые вычисляются через  $\nu_l$  – частоты взаимных столкновений между молекулами одного сорта.

В случае использования полностью макроскопического подхода к решению конкретных газодинамических задач, кинетические коэффициенты и параметры уравнений состояния (УРС) задаются в виде аппроксимаций экспериментальных данных или теоретических моделей. Например, данные для воздушной смеси можно рассчитать на основе таблиц из справочников [50 – 52]. В качестве компонент скоростей  $\mathbf{u}_l^*$  и энергии  $E_l^*$  можно использовать аппроксимирующие зависимости, удовлетворяющие условиям термодинамического равновесия смеси.

В случае использования многомасштабного подхода (см. ниже), кинетические коэффициенты чистых газов и параметры УРС выбираются из

базы данных молекулярных расчетов (БДРМ), которая сформирована и постоянно пополняется в рамках развития программного комплекса. При этом компоненты скоростей  $\mathbf{u}_l^*$  и энергии  $E_l^*$  вычисляются с помощью следующих формул:

$$\begin{aligned} \mathbf{u}_l^* &= \frac{1}{N_l} \sum_{i=1}^{N_l} \mathbf{v}_{l,i}^*, & \mathbf{u}^* &= \sum_{l=1}^N m_l N_l \mathbf{u}_l^* / \sum_{l=1}^N m_l N_l; \\ \mathbf{u}_{l,T}^* &= \mathbf{u}_l^* - \mathbf{u}^*, & T_l^* &= \frac{|\mathbf{u}_{l,T}^*|^2}{\mathfrak{R}_l}, & T^* &= \sum_{l=1}^N n_l T_l^* / \sum_{l=1}^N n_l; \\ p_l^* &= \rho_l Z_l^* \mathfrak{R}_l T_l^*, & p^* &= \rho Z^* \mathfrak{R} T^*; \\ E_l^* &= \rho_l \left( \frac{|\mathbf{u}_l^*|^2}{2} + c_{v,l}^* T_l^* \right), & E^* &= \sum_{l=1}^N E_l^*. \end{aligned} \quad (1.11)$$

Скорости частиц  $\mathbf{v}_{l,i}^*$ , факторы сжимаемости для газовых компонент  $Z_l^*$  и для смеси  $Z^*$ , удельная теплоемкость при постоянном объеме  $c_{v,l}^*$  определяются на основе непосредственных подсчетных вычислений по уравнениям молекулярной динамики (см. п. 1.1.5).

### 1.1.2 Модель теплопроводности

Для отражения теплопроводных эффектов в рамках рассматриваемой КГид модели течения система уравнений (1.1) – (1.3) дополняется выражением:

$$\frac{\partial T}{\partial t} = \nabla \left[ \frac{1}{\text{RePr}} \nabla T + (\mathbf{w} - \mathbf{u}) T \right], \quad (1.12)$$

где  $\text{Pr} = \frac{\eta c_p}{\kappa}$  – число Прандтля,  $c_p$  – удельная теплоемкость при постоянном давлении,  $\kappa$  – коэффициент теплопроводности.

Уравнения (1.1) – (1.3) дополняются членами:

$$\mathbf{Gr}T, \nabla \mathbf{Gr}T \text{ и } -\tau \mathbf{Gr}T \quad (1.13)$$

соответственно, тут  $\mathbf{Gr} = \{Gr_x, Gr_y, Gr_z\}^T$ . При учете силы тяжести, действующей вдоль оси OY,  $\mathbf{Gr} = \{0, Gr, 0\}^T$ , где  $Gr = \frac{g \rho^2 L_h^3 \beta \Delta T}{\eta^2}$  - число Грасгофа,  $g$  – ускорение свободного падения,  $\beta$  – коэффициент объемного расширения среды,  $\Delta T$  – характерный перепад температур.

Итоговая система уравнений вязкой несжимаемой жидкости включает уравнения (1.1) - (1.4), (1.12), (1.13) и дополняется граничными и начальными условиями для скоростей, давления и температуры.

### 1.1.3 Модель эволюции концентрации

При моделировании поведения примесей, содержащихся в потоке жидкости или газа, на макроскопическом уровне вводились концентрации ее компонент. Для получения их распределения и эволюции в исследуемой области очистной системы для учета конвекционно-диффузионного движения компонент примеси использовались классические уравнения конвекции-диффузии, дополненные КГид регуляризацией. Эти уравнения в безразмерной форме имеют следующий вид:

$$\frac{\partial C_k}{\partial t} = \nabla \left[ D_k \nabla C_k + (\mathbf{w} - \mathbf{u}) C_k \right], \quad (1.14)$$

здесь  $C_k$  – концентрация частиц (ионов) сорта  $k$ ,  $D_k = \frac{D_{0,k}}{u_0 L_h}$  – безразмерный коэффициент диффузии частиц.

Система уравнений, состоящая из  $n_c$  (количество компонент примеси) уравнений, (1.14) дополняется соответствующими задаче граничными условиями и начальными условиями.

### 1.1.4 Электростатическая модель

Моделирование электромагнитных фильтров требует учет силы Лоренца, в уравнении (1.14) посредством введения члена:

$$-\nabla(F_k \mathbf{F} C_k), \quad (1.15)$$

где  $F_k = \frac{q_k \mu_{\text{э}k} E_0}{u_0}$  – коэффициент нормировки силы Лоренца,  $q_k$  – заряд

частиц в относительных единицах с учетом знака,  $\mu_{\text{э}k}$  – коэффициент

подвижности частиц,  $E_0 = \frac{\varphi_0}{L_h}$  – константа нормировки напряженности,

$$\varphi_0 = \frac{\max_k (|q_k C_{0,k}|) L_h^2}{\varepsilon} \text{ – константа нормировки потенциала, } \mathbf{F} = \mathbf{E} + B_0 \left[ \mathbf{u} \times \frac{\mathbf{B}}{|\mathbf{B}|} \right]$$

– сила Лоренца,  $\mathbf{E}$  – напряженность электрического поля,  $B_0 = \frac{u_0}{E_0} |B|$  –

константа нормировки магнитной индукции,  $\varepsilon$  – диэлектрическая проницаемость среды,  $\mathbf{B}$  – вектор магнитной индукции.

Для определения напряженности будем использовать потенциальную постановку:

$$\nabla(\nabla \varphi) = - \sum_k^{n_c} \frac{q_k C_{0,k}}{\max_k (|q_k C_{0,k}|)} C_k, \quad (1.16)$$

$$\mathbf{E} = -\nabla \varphi, \quad (1.17)$$

где  $\varphi$  – потенциал электрического поля.

Исследуемым в диссертационной работе фильтрационным процессам характерно магнитное число Рейнольдса много меньше единицы. Этот факт дает возможность считать магнитное поле в системе внешним фактором. Его эволюция в данном случае не учитывается, оно инициируется либо постоянным сильным магнитом, либо соленоидом, по которому течет большой постоянный круговой ток.

Уравнения (1.16), (1.17) дополняются граничными и начальными условиями, соответствующими постановке задачи.

### 1.1.5 Микромасштабная модель

В рамках диссертации микромасштабная модель используется в рамках многомасштабного подхода (см. п. 1.1.6). Она реализуется с помощью разработанных ранее программных модулей комплекса GIMM\_NANO, в том числе GIMM\_APP\_MD\_CPU\_Gas\_Metal [53], KIAM\_MolSDAG\_CPU [54]. Далее кратко приведем основы используемых в рамках данного ПО численных подходов.

Для моделирования обменных членов в потоке, а также процессов в пограничных слоях на микроскопическом уровне используются уравнения классической динамики Ньютона [55, 56]. Приведем эти уравнения для случая многих типов частиц:

$$m_l \frac{d\mathbf{v}_{l,i}}{dt} = \mathbf{F}_{l,i}, \quad \mathbf{v}_{l,i} = \frac{d\mathbf{r}_{l,i}}{dt}, \quad i = 1, \dots, N_l, \quad l = 1, \dots, L., \quad (1.18)$$

где  $i$  – номер частицы,  $l$  – тип частиц,  $N_l$  – полное число частиц типа  $l$ ,  $m_l$  – масса частиц типа  $l$ ,  $\mathbf{r}_{l,i} = (r_{x,l,i}, r_{y,l,i}, r_{z,l,i})$  и  $\mathbf{v}_{l,i} = (v_{x,l,i}, v_{y,l,i}, v_{z,l,i})$  – радиус-вектор и вектор скорости  $i$ -й частицы типа  $l$ ,  $\mathbf{F}_{l,i} = (F_{x,l,i}, F_{y,l,i}, F_{z,l,i})$  – суммарная сила, действующая на данную частицу.

Сила  $\mathbf{F}_{l,i}$ , действующая на выделенную частицу  $i$  типа  $l$ , представляет собой сумму сил взаимодействия  $i$ -й частицы с ее окружением и силы внешнего воздействия:

$$\mathbf{F}_{l,i} = -\frac{\partial U(\mathbf{r}_{l,1}, \dots, \mathbf{r}_{l,N_l})}{\partial \mathbf{r}_{l,i}} + \mathbf{F}_{l,i}^{ext}, \quad i = 1, \dots, N_l, \quad l = 1, \dots, L., \quad (1.19)$$

здесь  $U$  – суммарная потенциальная энергия системы частиц,  $\mathbf{F}_{l,i}^{ext}$  – сила взаимодействия  $i$ -й частицы типа  $l$  с внешней средой.

Отметим, что эффекты вращения частиц в рассматриваемом диапазоне параметров не являются существенными и не учитываются.

Потенциальная энергия системы представляется в виде суммы

парциальных энергий, вычисление которых происходит согласно формуле выбранного потенциала взаимодействия.

В случае, когда рассматриваются взаимодействия между атомами и/или молекулами система (1.18), (1.19) совпадает с уравнениями молекулярной динамики. В случае, когда речь идет об атомно-молекулярных кластерах или крупных частицах, система (1.18), (1.19) является частью общего метода частиц, в котором могут присутствовать дополнительные уравнения и связи, например, уравнения для моментов импульса частицы. Фактически, разница между этими вариантами систем уравнений определяется моделью частицы и видом потенциалов взаимодействия. Здесь важно определить, является ли частица реальной или виртуальной, имеет ли она жесткую пространственную конфигурацию или является эластичной, а также, какие механизмы заложены в потенциал взаимодействия частиц.

Для компонент воздушной смеси на молекулярном уровне в данной работе использовались потенциалы Леннарда-Джонса, для молекул загрязнителей и материала гранул использовались потенциалы Ми. Потенциалы взаимодействия частиц разных сортов определялись в рамках модели Леннарда-Джонса на основе правил Лоренца-Бершелота [57]. В качестве начальных условий принимались равновесные состояния системы, сконструированные при помощи характеристик потока, полученных в макрорасчете.

Для компонент водной среды молекулярный уровень не использовался. Однако на поверхности электродов и на поверхности гранул сорбента необходимо было вычислить коэффициенты сорбции частиц загрязнителя. Для этого применялась модель молекулярной динамики. В предварительном расчете рассматривался небольшой фрагмент поверхности электрода или гранулы, в рамках которого вычислялось взаимодействие частиц загрязнителя и атомов поверхности. Для замыкания задачи использовались периодические граничные условия. После проведения расчета до установления термодинамического равновесия макропараметры

молекулярной системы, в том числе коэффициент сорбции, определялись в соответствии с [57 – 60].

### **1.1.6 Многомасштабный подход, связь микро- и макромасштабов**

Опишем многомасштабный подход, использованный в настоящей работе. Он является комбинацией упрощенного и истинного многомасштабных подходов. Эта комбинация позволяет при необходимости уточнять те или иные физические процессы, протекающие на макроуровне, с помощью применения моделей микроскопического уровня. Она предопределила выбор конкретных математических моделей, использованных в работе.

Поскольку речь идет об анализе физических процессов на этапе тонкой фильтрации, то вследствие малого размера частиц загрязнителей и малой их общей массы по сравнению с массой среды в сечении очистной системы их обратное воздействие на поток в целом является несущественным. Этот факт позволяет, однажды рассчитав макроскопические характеристики потока, использовать полученные значения скорости и давления для последующих стадий моделирования.

С помощью макропараметров потока возможно посредством применения методов частиц и методов молекулярной динамики, получить равновесные коэффициенты сорбции загрязнителя, реализующиеся на твердых стенках фильтрующего резервуара и сорбционных элементов (гранул сорбента). Поскольку данный этап моделирования является вычислительно емким, его можно провести независимо для отдельных диапазонов давлений, температур и скоростей среды, а также для разных ее составов. Накопление такой информации в базе данных молекулярных расчетов позволит в будущем применять по возможности упрощенный многомасштабный подход.

Возвращаясь к макромоделю, возможно получить распределение концентрации в рассматриваемой области в целом посредством учета

конвекционно-диффузионных, электромагнитных и сорбционных воздействий на концентрацию примеси. В дальнейшем возможно продолжить моделирование на макроуровне, если воздействие изменения концентрации не приводит к существенному изменению параметров захвата. Иначе, необходимо проводить повторное микромоделирование для актуализации сорбционных коэффициентов.

Таким образом, в предлагаемой версии многомасштабного подхода предполагается либо чередование макроскопических и микроскопических вычислений (истинно многомасштабный подход), реализуемое посредством метода расщепления по физическим процессам и масштабам, либо замена микроскопических вычислений параметрами из БДРМ.

Истинно многомасштабный подход (ИМП) подробно описан в работах [61, 62] и не является целью настоящей диссертации. Тем не менее, приведем некоторые детали этого подхода.

Во-первых, ИМП предполагает чередование на каждом шаге интегрирования динамической модели вычислений макроскопических уравнений на сетке и подсеточных вычислений по уравнениям динамики частиц. В частности, для газовой среды это означает решение КГД системы и уравнений молекулярной динамики.

Во-вторых, между вышеуказанными вычислениями осуществляется переход от макропараметров газа к системам атомов и молекул либо во всех ячейках сетки, либо только в пограничных слоях. Для этого используются следующие предположения:

- 1) в каждом микрообъеме вдали от стенок и твердых частиц сохраняются суммарные плотность, импульс и энергия смеси;
- 2) при столкновениях молекул газа между собой взаимодействие определяется соответствующими потенциалами для газовых компонент смеси (например, потенциалами Леннарда-Джонса [63]);

3) при столкновениях газа со стенкой взаимодействия определяются комбинацией разных потенциалов, например, Леннарда-Джонса и Морзе [64];

4) взаимодействия частиц металла между собой определяется потенциалом погруженного атома [65];

5) энергия газа может поглощаться стенкой или наоборот стенка нагревает газ.

Переход в конкретной ячейке сетки (или контрольном объеме)  $V_m$ , например, для бинарной смеси газов, схематически можно представить следующим образом:

– общее число молекул:  $V_m \rightarrow N = (n_a + n_b)V_m$ ;

– координаты скорости и молекул:  $\mathbf{u}_a, \mathbf{u}_b \rightarrow \mathbf{r}_{a,i}, \mathbf{r}_{b,j}, \mathbf{c}_{a,i}, \mathbf{c}_{b,j}$ ;

– средние скорости молекул:  $\mathbf{u}_l \equiv \langle \mathbf{c}_l \rangle = \frac{1}{N_l} \sum_{i=1}^{N_l} \mathbf{c}_{l,i}, l = a, b$

При этом предполагается:

1) распределение молекул по ячейке считается равномерным и таким, чтобы суммарная потенциальная энергия молекул совпадала с внутренней энергией микрообъёма;

2) распределение модуля скорости молекул каждого сорта считается локально максвелловским, так что суммарная кинетическая энергия молекул совпадает с кинетической энергией микрообъёма;

3) Распределение направлений векторов тепловых скоростей молекул считается равномерным и таким, чтобы суммарный импульс молекул совпадал со средним импульсом микрообъёма.

В результате перехода от макропараметров к микросистеме частиц получается начальное условие для расчета их эволюции с существенно меньшим временным шагом (обычно на три порядка меньшим, чем шаг интегрирования КГД системы). Цель этих подсеточных вычислений состоит

в расчете обменных членов и истинных потоков газа на границах газовой среды.

В-третьих, после завершения подсеточных вычислений (критерием их останова является установление локального термодинамического равновесия в каждом объеме  $V_m$ ) пересчитываются макропараметры газа в соответствующих ячейках сетки. Для бинарной смеси это выглядит следующим образом:

$$\mathbf{c}_{l,i} \rightarrow \bar{\mathbf{c}}_{l,i} \rightarrow \langle \bar{\mathbf{c}}_l \rangle = \frac{1}{N_l} \sum_{i=1}^{N_l} \bar{\mathbf{c}}_{l,i} \equiv \bar{\mathbf{u}}_l, \quad l = a, b; i = 1 \dots N_l;$$

$$\bar{\mathbf{u}} = \frac{m_a n_a \bar{\mathbf{u}}_a + m_b n_b \bar{\mathbf{u}}_b}{m_a n_a + m_b n_b}; \quad \bar{\mathbf{u}}_{l,T} = \bar{\mathbf{u}}_l - \bar{\mathbf{u}};$$

$$S_a^u = \nu_{ab} \rho_a (\bar{\mathbf{u}}_a - \mathbf{u}_a), \quad S_b^u = \nu_{ba} \rho_b (\bar{\mathbf{u}}_b - \mathbf{u}_b);$$

$$\bar{T}_l = \frac{|\bar{\mathbf{u}}_{l,T}|^2}{i_l \mathfrak{R}_l}, \quad \bar{T} = \frac{n_a i_a \bar{T}_a + n_b i_b \bar{T}_b}{n_a i_a + n_b i_b}; \quad \bar{p}_l = Z_l \rho_l \mathfrak{R}_l \bar{T}_l;$$

$$\bar{p} = \bar{p}_a + \bar{p}_b, \quad \bar{E}_l = \frac{\rho_l |\bar{\mathbf{u}}_l|^2}{2} + \rho_l c_{v,l} T_l, \quad \bar{E} = \bar{E}_a + \bar{E}_b;$$

$$S_a^E = \nu_{ab} (\bar{E}_a - E_a), \quad S_b^E = \nu_{ba} (\bar{E}_b - E_b).$$

Упрощенный многомасштабный подход напротив разработан в рамках данного исследования. Его целью является определение средствами микромасштабных моделей следующих коэффициентов и параметров:  $A_k$ ,  $C_k^{**}$ ,  $D_k$ . Последние являются результатами осреднения данных о взаимодействии конкретных молекулярных систем. Методики получения этих параметров аналогичны указанным в работах [66 – 68].

### 1.1.7 Пересчет к размерным переменным

Описанные в рамках пунктов 1.1.1 – 1.1.4 модели представлены в безразмерной форме, поэтому перед непосредственным сравнением результатов необходимо производить процедуру пересчета рассчитываемых

величин в требуемую систему единиц измерения в соответствии с результатами теории подобия [69, 70].

В качестве определяющего критерия нормировки модели течения 1.1.1 будем использовать число Рейнольдса. Из выражения  $Re = \frac{\rho u_0 L_h}{\eta}$  видно, что  $L_h$ , соответствующее геометрическому параметру расчетной области,  $\eta$  и  $\rho$ , характеризующие среду, являются свободными параметрами. Данный факт, при закреплении числа Рейнольдса, позволяет получить выражение для расчета характерной скорости потока:

$$\tilde{\mathbf{u}} = u_0 \mathbf{u} = \frac{Re \eta}{\rho_0 L_h} \mathbf{u}, \quad (1.20)$$

где  $\tilde{\mathbf{u}}$  - скорость в размерных единицах. Выражение для пересчета давления примет следующий вид (из того, что  $\frac{p_0}{\rho_0 U_0^2} = 1$ ):

$$\tilde{p} = p_0 p = \frac{Re^2 \eta^2}{\rho_0 L_h^2} p. \quad (1.21)$$

где  $\tilde{p}$  - давление в размерных единицах. Размерное время вычисляется с помощью выражения:

$$\tilde{t} = t_0 t = \frac{\rho L_h^2}{Re \eta} t. \quad (1.22)$$

В модели 1.1.2 характеристикой вещества является число Прандтля. Отметим, что размерный коэффициент пересчета температуры  $T_0$  является свободным и задается непосредственно. Пересчет осуществляется очевидным образом:

$$\tilde{T} = T_0 T. \quad (1.23)$$

Коэффициенты концентрации компонент примесей модели 1.1.3 задается свободно, аналогично температурному. Формула пересчета:

$$\tilde{C}_k = C_{0,k} C_k. \quad (1.24)$$

При этом безразмерный коэффициент диффузии, посредством подстановки выражения для  $u_0$ :

$$D_k = \frac{D_{0,k}}{u_0 L_h} = \frac{D_{0,k} \rho_0}{\text{Re} \eta}. \quad (1.25)$$

Формулы пересчета потенциала и напряженности электрического поля для модели 1.1.4, при нормировке по концентрации и заряду примеси, имеют вид:

$$\tilde{\varphi} = \varphi_0 \varphi = \frac{\max_k (|q_k C_{0,k}|) L_h^2}{\varepsilon} \varphi, \quad \tilde{E} = E_0 E = \frac{\max_k (|q_k C_{0,k}|) L_h}{\varepsilon} E. \quad (1.26)$$

При подстановке выражения (U) и (E) в нормировочную константу силы Лоренца, получим выражение:

$$F_k = \frac{q_k \mu_{\text{ЭК}} \rho_0 L_h^2}{\eta \varepsilon \text{Re}} \max_k (|q_k C_{0,k}|). \quad (1.27)$$

Итоговый набор свободных параметров моделей имеет следующий вид:

- КГид модель: число Рейнольдса – Re;
- модель теплопроводности: нормировка температуры –  $T_0$ , числа Прандтля и Грасгофа – Pr, Gr;
- модель конвекции-диффузии примеси: коэффициент диффузии –  $D_k$ ;

– электростатическая модель: нормировочные константы силы Лоренца –  $F_k$ , нормировочная константа напряженности магнитного поля –  $B_0$

и набор отношений  $\frac{q_k C_{0,k}}{\max_k (|q_k C_{0,k}|)}$ .

Анализ предложенных моделей по критическим безразмерным параметрам показал, что они являются справедливыми как минимум в следующих диапазонах:  $Re = 1 \div 1000$ ,  $Pr = 1 \div 100$ ,  $Gr = 1 \div 100$ ,  $D_k = 0.001 \div 1$ ,  $F_k = 0 \div 100$ ,  $B_0 = 0 \div 100$ . Это позволяет сделать вывод о применимости построенных моделей к рассматриваемому кругу задач промышленной экологии.

## 1.2 Дополняющие соотношения

Одним из основных вопросов при рассмотрении моделей физических процессов является постановка корректных граничных и начальных условий. В данном пункте приведем набор данных дополняющих соотношений для каждой из использованных в рамках работы макромоделей.

### 1.2.1 Модель течения

Основной задачей при моделировании течения внутри фильтрационных устройств является задача о внутреннем напорном течении, при котором необходимо иметь граничные условия, описывающие входные и выходные отверстия, а также стенку.

Течение на входном отверстии может быть задано через известное распределение скорости и перепад давления. В рамках работы в основном будем использовать течение Пуазейля, определяемое следующими соотношениями, полученными из условия нормировки в центре потока

$$u \Big|_{r=0} = 1:$$

$$\mathbf{u}(r) = \left(1 - \frac{r^2}{R^2}\right) \mathbf{n}, \quad \frac{\partial p}{\partial n} = -\frac{2}{\text{Re} R^2}, \quad (1.28)$$

где  $R$  – радиус входного отверстия,  $\mathbf{n}$  – нормаль к поверхности отверстия.

На выходе будем использовать стандартные мягкие граничные условия, имеющие форму:

$$\frac{\partial \mathbf{u}}{\partial n} = 0, \quad p = 0, \quad (1.29)$$

Существуют различные варианты задания граничных условий в задачах внутреннего течения несжимаемой жидкости, основными предельными ситуациями являются прилипание и проскальзывание [71, 72]. При моделировании процессов фильтрации, на масштабах исследовательских и промышленных установок, наиболее корректными будут условия прилипания на стенке, имеющие следующий вид:

$$\mathbf{u} = \mathbf{0}, \quad \frac{\partial p}{\partial n} = 0. \quad (1.30)$$

Начальным состоянием будем считать покой среды:

$$\mathbf{u} = \mathbf{0}, \quad p = 0.5. \quad (1.31)$$

### 1.2.2 Модель теплопроводности

Граничные условия для теплопроводности требуют отражение процессов, происходящих на входных и выходных отверстиях, а также нагревательном элементе (при наличии) и стенках резервуара.

На входном и выходных отверстиях будем считать:

$$\frac{\partial T}{\partial n} = 0. \quad (1.32)$$

На стенке исследуемого объема допустим теплообмен:

$$\frac{\partial T}{\partial n} = \alpha(T - T_0), \quad (1.33)$$

где  $\alpha$  – коэффициент теплообмена,  $T_0$  – температура окружающей среды.

На нагревательном элементе необходимо учесть падение температуры при образовании солевого налета, для этого будем использовать следующее выражение:

$$\frac{\partial T}{\partial n} = F_0 + F_1 \frac{C_{1k} - C_k}{C_{1k}}, \quad (1.34)$$

где  $F_0$  – поток тепла при максимальном загрязнении,  $F_0 + F_1$  – поток при отсутствии загрязнения,  $C_{1k}$  – предельная захватываемая концентрация загрязняющего вещества.

В качестве начального состояния будем считать:

$$T = 0, \quad (1.35)$$

во всей области.

### 1.2.3 Модель эволюции концентрации

На входе в очистной резервуар граничные условия модели эволюции концентрации будем считать заданными непосредственно:

$$C_k = \frac{C_{0,k}}{\sum_k^{n_c} C_{0,k}}. \quad (1.36)$$

при условии равновесного выбора  $C_{0k}$ .

На входном и выходных отверстиях будем считать:

$$\frac{\partial C_k}{\partial n} = 0. \quad (1.37)$$

Пристеночные эффекты сорбции будем считать с помощью выражения:

$$\frac{\partial C_k}{\partial n} = A_k (C_k - C_k^*) \left( 1 - \frac{C_k}{C_k^{**}} \right), \quad (1.38)$$

где  $C_k^*$  – равновесное значение концентрации частиц  $k$ -го компонента примеси,  $C_k^{**}$  – максимальная концентрация  $k$ -го компонента, удерживаемая элементом поверхности гранулы сорбента,  $A_k$  – интенсивность захвата частиц  $k$ -го компонента загрязнителя поверхностью гранулы сорбента (может зависеть от общего числа вакансий на поверхности гранулы, свободных от всех наночастиц всех сортов загрязнителя).

Расчеты будем проводить от нулевых начальных условий:

$$C_k = 0. \quad (1.39)$$

#### 1.2.4 Электростатическая модель

В рамках электростатической модели достаточно ввести электронейтральные границы (стенка, входное и выходное отверстие) и электроды.

Используемые в рамках работы выражения имеют следующий вид:

- на электронейтральных границах:

$$\frac{\partial \varphi}{\partial n} = 0; \quad (1.40)$$

- на электродах:

$$\varphi = \varphi_k, \quad (1.41)$$

где  $\varphi_k$  – константное значение потенциала на  $k$ -ом электроде.

#### 1.3 Постановки задач

Рассматриваемые в данной работе задачи разделяются на три части: верификационные задачи проверки программной реализации численных

алгоритмов, валидационная задача моделирования комплексного процесса обессоливания воды и набор модельных задач промышленной экологии.

### 1.3.1 Задачи верификации

#### 1.3.1.1 Установление течения Пуазейля

Первой задачей верификации для проверки модели течения будем рассматривать установление течения Пуазейля в плоской прямоугольной области, изображенной на рис. 1.1.

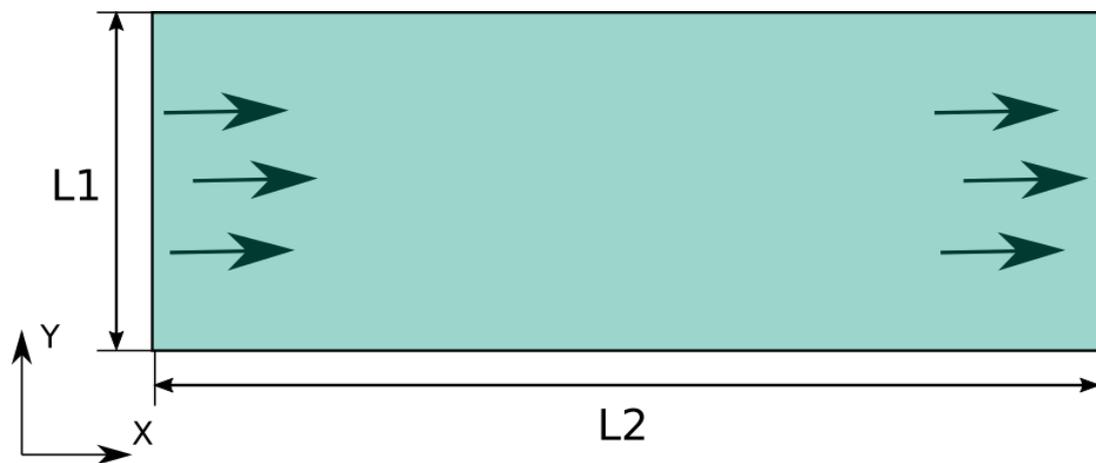


Рисунок 1.1 – Геометрия первой задачи верификации

В качестве размеров области примем  $L_1 = 1.0$  и  $L_2 = 4.0$ . Как известно, при установлении ламинарного течения решением данной задачи является перенос профиля Пуазейля вдоль оси  $X$  на всю область.

#### 1.3.1.2 Перенос концентрации

Для проверки модели конвекции-диффузии будем рассматривать задачу о переносе концентрации, распределенной в пространстве по закону Гаусса. Данный тест часто применяется для проверки корректности численных алгоритмов и их программной реализации [73]. Схема постановки изображена на рис. 1.2.

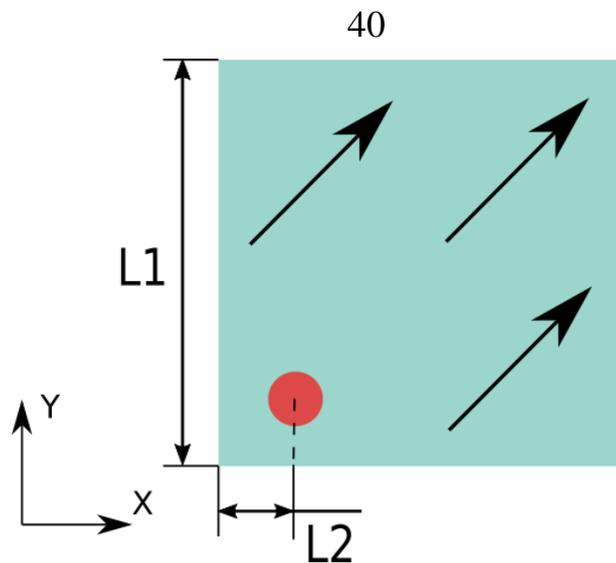


Рисунок 1.2 – Геометрия второй задачи верификации

Значение  $L_1$  примем равным 2.0,  $L_2$  равным 0.5. Проверка эволюции концентрации будет рассмотрена в соответствии с известным аналитическим решением.

### 1.3.1.3 Тепловое конвективное движение

Проверка термодинамической части будет проводиться с помощью моделирования вынужденного конвективного движения в квадратной области [43]. Условия задачи схематично представлены на рис. 1.3, красным цветом показана стенка с высокой температурой, синим – с низкой. Примем размер стороны квадрата  $L_1 = 1.0$ .

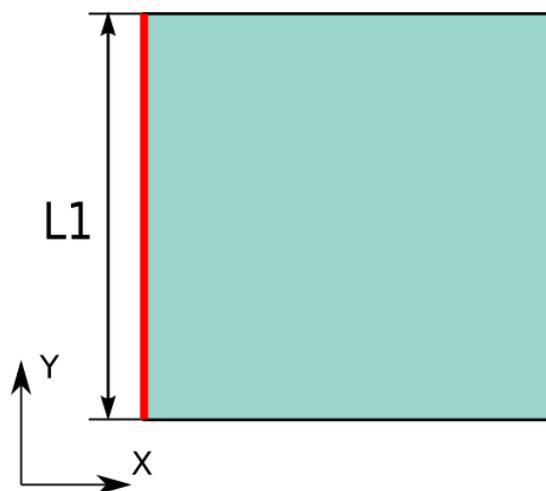


Рисунок 1.3 – Геометрия третьей задачи верификации

### 1.3.2 Задача валидации

Для валидации комплексной модели в целом, а также проверки электростатической части расчетного кода будет рассматриваться процесс обессоливания жидкости в постановке работ [74, 75]. Схема устройства приведена на рис. 1.4, красным цветом показан анод.

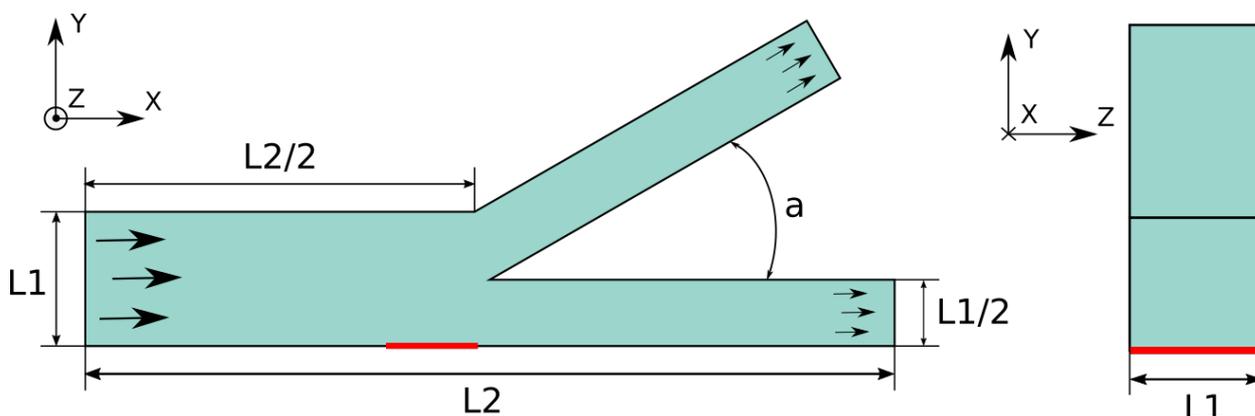


Рисунок 4 – Геометрия валидационной задачи

Размеры расчетной области в соответствии с обозначениями рис. 1.4 имеют следующие значения:  $L_1 = 100$  мкм,  $L_2 = 5000$  мкм,  $a = 30^\circ$ .

### 1.3.3 Модельные задачи

В настоящей работе были выбраны актуальные для математического анализа и практического применения задачи фильтрации, соответствующие требованиям технологий замкнутого цикла и обеспечивающие высокую контролируемую степень очистки.

#### 1.3.3.1 Задача об электромагнитной очистке водной среды

При рассмотрении задач очистки водной среды интересной оказалась фильтрация с помощью электромагнитного воздействия. В практическом плане ее преимущество состоит в простоте конструкции и технического обслуживания установки, широком диапазоне и простой настройке режимов очистки. В научном плане здесь еще остается широкое поле для исследований, связанных с геометрическими особенностями существующих

и будущих очистных систем, то есть имеется запрос на повышение точности проектных расчетных исследований.

Схема вычислительного эксперимента представлена на рис. 1.5. Расчетная область имеет одно входное отверстие в левой части и два выходных, первое из которых находится в нижней части области (на рис. 1.5 обозначено бирюзовым цветом), второе расположено справа. Радиусы отверстий одинаковы и равны  $R_1 = 0.5$ . Радиус расширения  $R_2 = 1.0$ . Характерные длины расчетной области:  $L_1 = 6.0$ ,  $L_2 = 1.5$ ,  $L_3 = 3.0$ . Вектор магнитной индукции  $\mathbf{B}$  направлен вдоль оси  $Z$ . В качестве рассчитываемых параметров примем скорость и давление внутри области, а также объемную концентрацию примеси при установившемся течении.

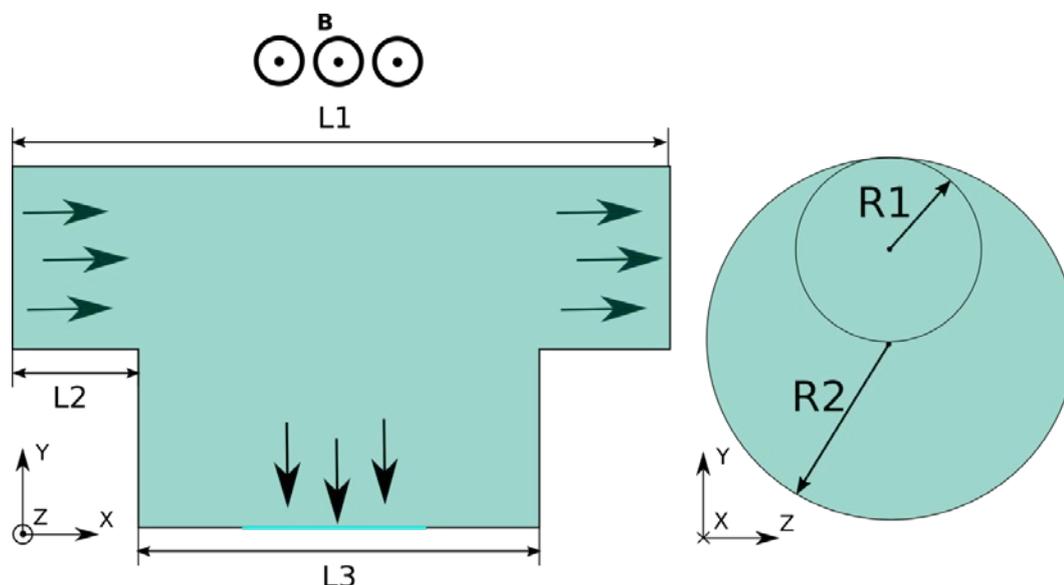


Рисунок 1.5 – Геометрия первой модельной задачи

### 1.3.3.2 Задача о сорбционной очистке воздушной среды

В случае фильтрации воздушной среды, наиболее соответствующей технологиям замкнутого цикла, является методика очистки посредством гранулированных сорбентов. С практической точки зрения здесь имеются все основания, считать данную технологию безотходной, при наличии возможности полной регенерации сорбента. Однако в научном плане эта технология требует широкомасштабных исследований. В текущей работе

исследуем процесс загрязнения сорбционной области, приводящий к падению эффективности очистки.

Для этого рассмотрим воздушный фильтр, представляющий из себя цилиндрическую трубу, содержащую в центре укладку гранулированных сорбентов, схема расчетной области приведена на рис. 1.6. Радиус трубы примем  $R_1 = 1$ , длина  $L_1 = 12$ . Сорбционная область будет содержать 210 сферических гранул радиуса 0.15. В данной задаче будем рассчитывать течение внутри области, эволюцию объемной концентрации примеси и поверхностной концентрации на сорбционных элементах.

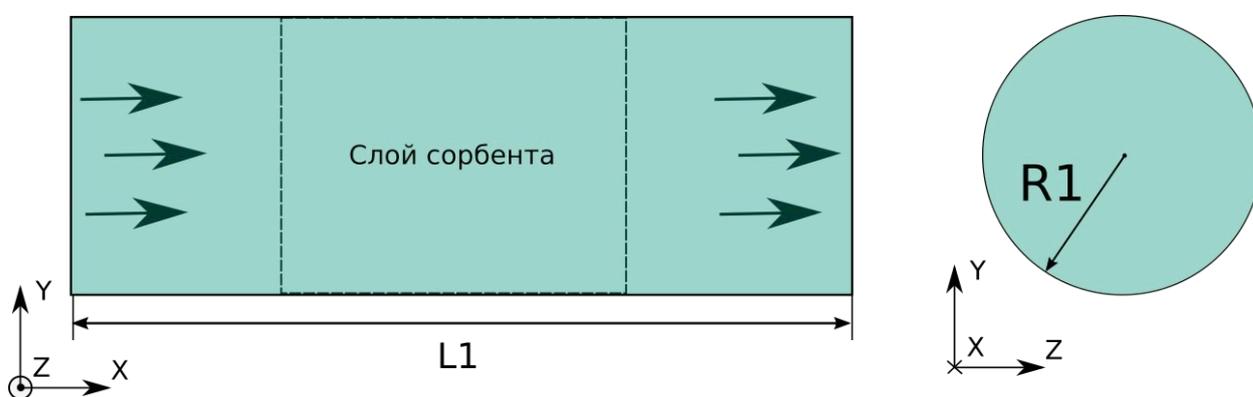


Рисунок 1.6 – Геометрия второй модельной задачи

### 1.3.3.3 Задача о загрязнении тепловыделяющего элемента

При длительном периоде эксплуатации тепловыделяющего элемента (ТЭН) в условиях жесткой воды, эффективность его работы существенно падает. Это связано в первую очередь с образованием накипи, получаемой из прокачиваемого раствора вследствие падения растворимости солей с увеличением температуры среды. Падение эффективности ТЭН приводит к увеличению энергопотребления, а в предельном случае к выходу из строя данного элемента [76]. Для предотвращения подобных ситуаций, а также установления циклов эксплуатации-регенерации необходимо иметь возможность моделирования процесса образования накипи.

Будем рассматривать нагревательную установку, геометрия которой изображена на рис. 1.7. Характерные размеры области, в соответствии с обозначениями рис. 1.7:  $L_1 = 4.5$ ,  $L_2 = 0.5$ ,  $L_3 = 3.5$ ,  $R_1 = 0.5$ ,  $R_2 = 3.0$ . ТЭН

представляет собой цилиндрическую трубку с радиусом 0.1 и габаритными размерами 1.5x2.0x0.2 по X, Y, Z соответственно.

Вычислительный эксперимент будем проводить в три стадии. В рамках первой стадии рассчитаем стационарные распределения скоростей, давления и температуры в области. На второй стадии проведем модельный расчет образования накипи в соответствии из жесткой воды, задаваемого следующей формулой:



Третья стадия будет заключаться в очистке нагревательного элемента от накипи прокачкой раствора соляной кислоты с формулой реакции:

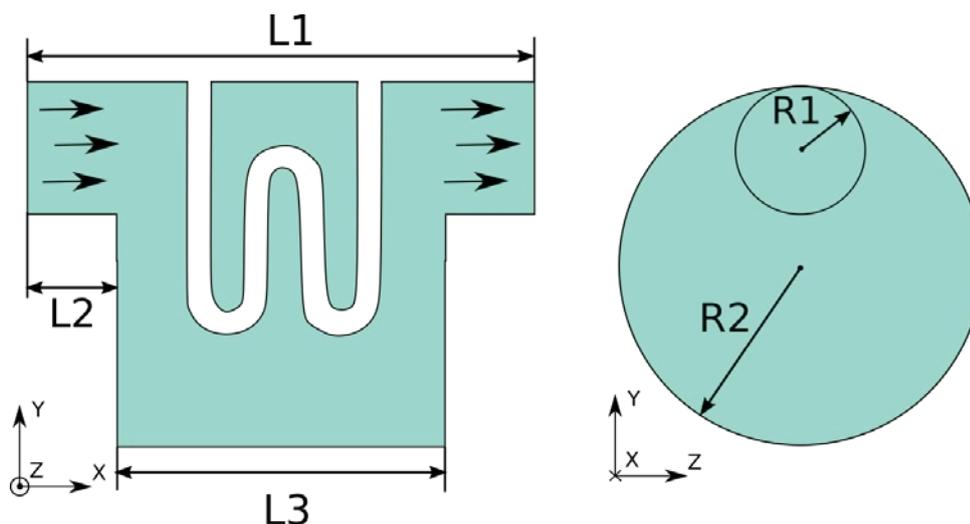


Рисунок 1.7 – Геометрия третьей модельной задачи

#### 1.4 Выводы по Главе 1

В первой главе обсуждались особенности моделирования процессов фильтрации водной и воздушной сред, на основании которых установлены необходимые модели макроуровня, включающие КГиД, модель теплопроводности, модель конвекции-диффузии компонент загрязнителя и модель электростатики. Был обсужден вопрос многомасштабного подхода, связывания микро- и макромасштабных моделей. Предложен набор задач,

состоящий из трех категорий: задачи верификации, задача валидации и модельные задачи промышленной экологии.

Основные результаты главы в части моделирования течения опубликованы в работах [25, 27, 28, 30], электромагнитной модели – [24, 26, 34], модели конвекции-диффузии с учетом эффектов сорбции – [32, 33], многомасштабного подхода – [29, 31].

## Глава 2. Численные методы и их реализация

### 2.1 Численные методы

Численная реализация отобранных для решения макроскопических уравнений механики сплошной среды базируется на методах расщепления по физическим процессам и сеточных методах конечного объема [77 – 80].

При аппроксимации пространственных операторов вышеуказанных эллиптических и параболических уравнений макромодели используются неструктурированные сетки, что определяется сложной реальной геометрией моделируемых систем очистки. В качестве контрольных объемов используются ячейки сетки. Все искомые величины усредняются по объему ячейки и определяются в ее центре, за исключением регуляризирующей поправки  $w$  в КГиД уравнениях, определяемой на гранях. Временные производные аппроксимируются явным образом. Данный подход позволяет использовать адаптивные неструктурированные сетки и достичь высокой степени аппроксимации границ даже геометрически сложной области.

Для численного расчета на микроуровне используется схема Верле [56] в скоростной форме. В перспективе наравне с ней могут быть использованы асимптотические методы интегрирования обыкновенных дифференциальных уравнений (см., например, [81]).

#### 2.1.1 Численные методы макроскопического уровня

Рассмотрим более подробно численные методы макроскопического уровня. Для этого введем сеточные параметры и обозначения на примере прямоугольных элементов, изображенных на рис. 2.1. Здесь  $V_i$  – площадь в двумерном и объем в трехмерном случае текущего  $i$ -ого сеточного элемента,  $P_i$  – точка центра текущего  $i$ -ого элемента,  $P_{ij}$  – точка центра  $j$ -соседнего к  $i$ -ому элемента,  $l_{ij}$  – расстояние между центрами соседних ячеек,  $\mathbf{n}_{ij}$  – единичный направляющий вектор от центра текущего элемента к центру соседнего,  $S_{ij}$  – направленные длина в двумерном случае и площадь в

трехмерном  $j$ -ой грани  $i$ -ого элемента, помноженная на внешнюю нормаль.

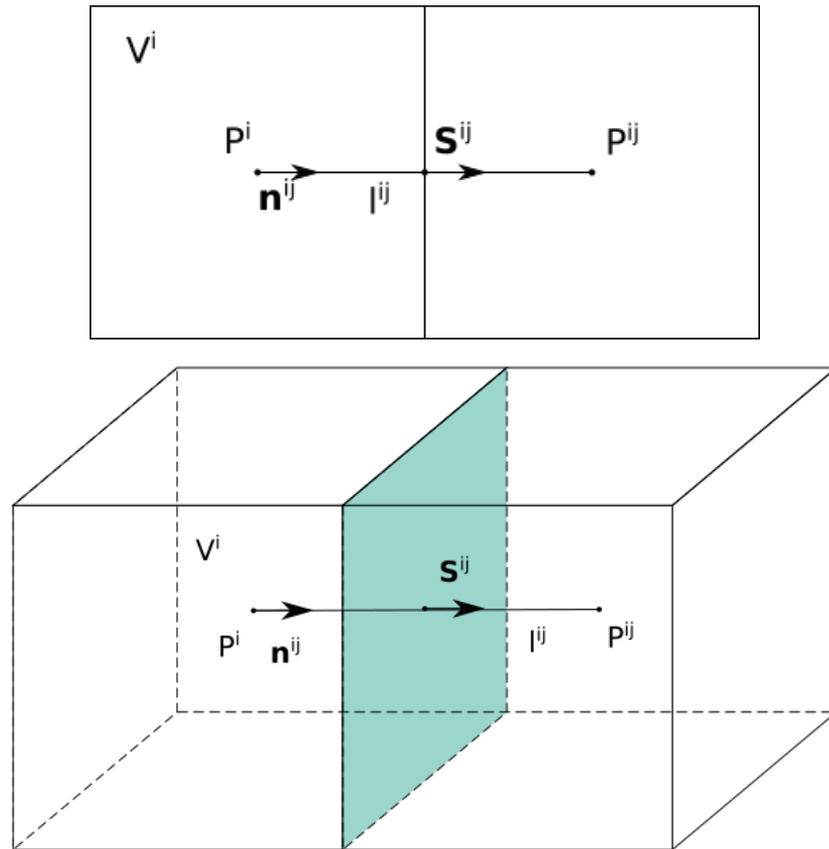


Рисунок 2.1 – Характерные сеточные величины для четырехугольных двумерных (сверху) и кубических трехмерных (снизу) конечных объемов

С учетом введенных обозначений сеточный аналог уравнений (1.1) – (1.3) примет вид:

$$\hat{\mathbf{u}}^i = \mathbf{u}^i + \frac{\tau_{time}}{V^i} \sum_{j=1}^m \left[ \frac{1}{\text{Re}} \left\{ \frac{\mathbf{n}^{ij}}{l^{ij}} \otimes (\mathbf{u}^{ij} - \mathbf{u}^i) + (\mathbf{u}^{ij} - \mathbf{u}^i) \otimes \frac{\mathbf{n}^{ij}}{l^{ij}} \right\} \cdot \mathbf{S}^{ij} + \left\{ \mathbf{w}^{ij} \otimes \bar{\mathbf{u}}^{ij} + \bar{\mathbf{u}}^{ij} \otimes \mathbf{w}^{ij} \right\} \cdot \mathbf{S}^{ij} - \left\{ \bar{\mathbf{u}}^{ij} \otimes \bar{\mathbf{u}}^{ij} \right\} \cdot \mathbf{S}^{ij} - \bar{p}^{ij} \mathbf{S}^{ij} \right], \quad (2.1)$$

$$\hat{p}^i = \left[ \sum_{j=1}^m \left( p^{ij} \frac{\mathbf{n}^{ij}}{l^{ij}} - \frac{1}{\tau} \bar{\mathbf{u}}^{ij} + (\bar{\mathbf{u}}^{ij}, \mathbf{n}^{ij}) \frac{\mathbf{u}^{ij} - \mathbf{u}^i}{l^{ij}}, \mathbf{S}^{ij} \right) \right] \left[ \sum_{j=1}^m \frac{(\mathbf{n}^{ij}, \mathbf{S}^{ij})}{l^{ij}} \right]^{-1}, \quad (2.2)$$

$$\mathbf{w}^{ij} = \tau \left[ (\bar{\mathbf{u}}^{ij}, \mathbf{n}^{ij}) \frac{\mathbf{u}^{ij} - \mathbf{u}^i}{l^{ij}} + \frac{p^{ij} - p^i}{l^{ij}} \mathbf{n}^{ij} \right], \quad (2.3)$$

где  $\hat{\mathbf{u}}^i$  – сеточный аналог скорости в  $i$  элементе на следующем временном

шаге,  $\mathbf{u}^i$  – аналог скорости в выделенном элементе на текущем временном шаге,  $\tau_{\text{time}}$  – шаг по времени,  $\bar{\mathbf{u}}^{ij} = 0.5(\mathbf{u}^{ij} + \mathbf{u}^i)$  – интерполяция скорости на центр грани сеточного элемента,  $\mathbf{u}^{ij}$  – сеточный аналог скорости в соседнем элементе,  $m$  – количество соседних к выделенному элементов (в случае треугольной сетки равное 3, квадратной сетки – 4, в случае кубической – 6),  $\bar{p}^{ij} = 0.5(p^{ij} + p^i)$  – интерполяция давления на центр грани сеточного элемента,  $\hat{p}^i$  – сеточный аналог давления в центре выделенной ячейки,  $p^{ij}$  – аналог давления в соседней ячейке,  $\mathbf{w}^{ij}$  – сеточный аналог регуляризирующей поправки.

Сеточные аналоги (2.1) – (2.3) дополняются естественным образом соответствующими граничными условиями (1.28) – (1.30). Расчет начинается с начальных условий (1.31). При этом вычисление одного временного шага скорости подразумевает последовательное вычисление выражений (2.1) – (2.3), выполняемое вплоть до установления стационарного режима течения. Отметим – в рамках временного шага скорости пересчет давления происходит единожды, что позволяет существенно ускорить расчет, при этом течение устанавливается по скорости и давлению совместно.

Уравнение модели теплопроводности (1.12) примут следующую разностную форму:

$$\hat{T}^i = T^i + \frac{\tau_{\text{time}}}{V^i} \sum_{j=1}^m \left( \frac{1}{\text{RePr}} \frac{(T^{ij} - T^i)}{l^{ij}} \mathbf{n}^{ij} + \bar{T}^{ij} (\mathbf{w}^{ij} - \bar{\mathbf{u}}^{ij}), \mathbf{S}^{ij} \right), \quad (2.4)$$

где  $\hat{T}^i$ ,  $T^i$  – температура в текущей  $i$  ячейке на следующем и текущем шагах по времени,  $T^{ij}$  – температура в соседней ячейке,  $\bar{T}^{ij} = 0.5(T^{ij} + T^i)$  – интерполяция температуры на грань сеточного элемента.

Разностные аналоги добавок (1.13) для выражений (2.1) – (2.3) примут вид:

$$\mathbf{Gr}T^i, \left[ \sum_{j=1}^m \bar{T}^{ij}(\mathbf{Gr}, \mathbf{S}^{ij}) \right] \left[ \sum_{j=1}^m \frac{(\mathbf{n}^{ij}, \mathbf{S}^{ij})}{l^{ij}} \right]^{-1} \text{ и } -\tau \mathbf{Gr}T^i. \quad (2.5)$$

Распространение компонент загрязнителя будем рассчитывать с помощью разностных аналогов уравнений конвекции-диффузии (1.14):

$$\hat{C}_k^i = C_k^i + \frac{\tau_{time}}{V^i} \sum_{j=1}^m \left( D_k \frac{C_k^{ij} - C_k^i}{l^{ij}} \mathbf{n}^{ij} + \bar{C}_k^{ij} (\mathbf{w}^{ij} - \bar{\mathbf{u}}^{ij}) \right), \quad (2.6)$$

где  $\hat{C}_k^i, C_k^i$  - концентрация k компонента примеси в i текущей ячейке на следующем и текущем временных шагах,  $C_k^{ij}$  - концентрация k компонента в соседней ячейке,  $\bar{C}^{ij} = 0.5(C^{ij} + C^i)$  - интерполяция концентрации k компоненты на грань сеточного элемента.

Эволюция концентрации рассчитывается покомпонентно посредством выражений (2.6), с учетом граничных условий (1.36) – (1.38), от начальных условий (1.39) при моделировании сорбционной фильтрации. При проведении численного эксперимента по задачам электромагнитной очистки набор уравнений дополняется моделью электростатики.

Электростатическая модель (1.16), (1.17) в разностной форме будет иметь следующий вид:

$$\hat{\phi}^i = \left[ V^i \sum_{k=1}^p \frac{q_k C_{k0}}{\max_k (|q_k C_{k0}|)} C_k^i + \sum_{j=1}^m \frac{\phi^{ij}}{l^{ij}} (\mathbf{n}^{ij}, \mathbf{S}^{ij}) \right] \left[ \sum_{j=1}^m \frac{(\mathbf{n}^{ij}, \mathbf{S}^{ij})}{l^{ij}} \right]^{-1}, \quad (2.7)$$

$$\hat{\mathbf{E}}^i = -\frac{1}{V^i} \sum_{j=1}^m \bar{\phi}^{ij} \mathbf{S}^{ij}, \quad (2.8)$$

где  $\hat{\phi}^i, \hat{\mathbf{E}}^i$  - значения потенциала и напряженности электрического поля на следующей итерации в i ячейке,  $\phi^i, \phi^{ij}$  - значение потенциала в текущей i ячейке и j соседней,  $\bar{\phi}^{ij} = 0.5(\phi^{ij} + \phi^i)$  - интерполяция потенциала на грань элемента.

Силу Лоренца будем определять на гранях сеточных элементов как:

$$\mathbf{F}^{ij} = \bar{\mathbf{E}}^{ij} + B_0 \left[ \bar{\mathbf{u}}^{ij} \times \mathbf{V}^{ij} \right], \quad (2.9)$$

где  $\bar{E}^{ij} = 0.5(E^{ij} + E^i)$  - интерполяция напряженности на грань сеточного элемента,  $\mathbf{V}^{ij}$  – вектор магнитной индукции на грани ячейки.

Дискретный аналог добавки (1.15) к выражению (2.6) примет вид:

$$-\frac{\tau_{time}}{V^i} F_k \sum_{j=1}^m \left( \bar{C}_k^{ij} \mathbf{F}^{ij}, \mathbf{S}^{ij} \right). \quad (2.10)$$

Выражения (2.7) – (2.8) последовательно вычисляются для каждого временного шага при расчете концентрации единойжды вплоть до искомого значения времени с учетом граничных условий (1.40), (1.41).

### 2.1.2 Численные методы микроскопического уровня

Численное решение уравнений молекулярной динамики реализуется посредством использования алгоритма Верле в скоростной форме, принимающего следующий вид:

$$\mathbf{F}_{l,i}^0 = \mathbf{F}_{l,i} \left( \mathbf{r}_{1,1}^0, \dots, \mathbf{r}_{1,N_1}^0; \dots; \mathbf{r}_{N_{tot},1}^0, \dots, \mathbf{r}_{N_{tot},N_{N_{tot}}}^0 \right), \quad i = 1, \dots, N_l, \quad l = 1, \dots, N_{tot}, \quad (2.11)$$

$$\forall n = 0, 1, 2, \dots: \quad \mathbf{r}_{l,i}^{n+1} = \mathbf{r}_{l,i}^n + \mathbf{v}_{l,i}^n \Delta t + \frac{\mathbf{F}_{l,i}^n (\Delta t)^2}{m_l}, \quad i = 1, \dots, N_l, \quad l = 1, \dots, N_{tot}, \quad (2.12)$$

$$\mathbf{F}_{l,i}^{n+1} = \mathbf{F}_{l,i} \left( \mathbf{r}_{1,1}^{n+1}, \dots, \mathbf{r}_{1,N_1}^{n+1}; \dots; \mathbf{r}_{N_{tot},1}^{n+1}, \dots, \mathbf{r}_{N_{tot},N_{N_{tot}}}^{n+1} \right), \quad i = 1, \dots, N_l, \quad l = 1, \dots, N_{tot}, \quad (2.13)$$

$$\mathbf{v}_{l,i}^{n+1} = \mathbf{v}_{l,i}^n + \frac{\mathbf{F}_{l,i}^{n+1} + \mathbf{F}_{l,i}^n}{2m_l} \Delta t, \quad i = 1, \dots, N_l, \quad l = 1, \dots, N_{tot}, \quad (2.14)$$

где  $\mathbf{F}_{l,i}^n$  – вектор силы на шаге n,  $\mathbf{F}_{l,i}$  – процедура расчета сил.

Эволюция системы получается разрешением алгебраической системы (2.11) – (2.14) с учетом периодических граничных условий от равновесного состояния системы, задаваемого параметрами, полученными в результате макромасштабной модели.

Иначе, алгебраическая система (2.11) – (2.14) разрешается независимо с некоторым шагом начальных параметров равновесного состояния. Посредством обработки полученных распределений строятся зависимости используемых в рамках макромодели коэффициентов в форме таблицы. Данные таблицы в дальнейшем применяются для определения коэффициентов при макрорасчетах по рассчитанным макропараметрам посредством линейной интерполяции значений.

### **2.1.3 Переход от макроскопического уровня к микроскопическому и обратно**

В рамках реализации истинно многомасштабного подхода требуется на этапах схемы расщепления по физическим процессам осуществлять переход от макроскопических параметров среды к конкретным ансамблям атомов и молекул. На последующих этапах требуется вычисление макроскопических величин по параметрам атомно-молекулярной системы.

Переход от макроскопического уровня к микроскопическому частично описан в п. 1.1.6. Он осуществляется в предположении установления локального термодинамического равновесия внутри выделенного контрольного объема или ячейки сетки. Относительно простым случаем является такой переход вдали от границ расчетной области, которые являются обычно твердыми поверхностями (например, металлическими). Для газовой среды предполагается, что функция распределения модуля скорости частиц смеси (атомов или молекул) является локально-максвелловской, отвечающей средней температуре смеси в объеме. Среднее направление скоростей частиц в объеме совпадает с направлением средней скорости среды. Пространственное распределение частиц в объеме считается однородным.

Более сложным случаем является переход в пограничном слое газ-металл, где к частицам газа добавляется атомно-молекулярная система металлического типа, совершающая колебательные движения в соответствии

с температурой стенки. В этом случае объем разделяется на две части, в каждой из которых реализуется локальное термодинамическое равновесие. Затем производится молекулярный расчет, целью которого является установление равновесия во всем объеме и передача импульса и энергии от газа к металлу или наоборот.

Обратный переход осуществляется по правилам статистической физики. То есть средние величины скорости, температуры и давления переносятся на макроскопический уровень.

Подробно вышеуказанные переходы обсуждались в [61].

## **2.2 Программная реализация**

Программная реализация описанных в п. 2.1.1 методов расчета и алгоритмов основывается на требованиях расширяемости вычислительного ядра, допускающего при необходимости введение дополнительных типов сеток, численных схем, моделей и алгоритмов. Кроме того, разрабатываемое программное обеспечение должно поддерживать возможности эффективного параллельного счета, в том числе на персональных вычислительных станциях, кластерах и суперкомпьютерах. Сегодня, для формирования подобных программных комплексов, стандартным стал подход реализации объектно-ориентированной библиотеки классов.

Итоговый список основных требований к вычислительному ядру представляет собой следующее:

- наличие общего подхода ввода-вывода, в том числе параллельного;
- возможность проведения вычислений на неструктурированных сетках, состоящих из различных типов сеточных элементов (треугольники, четырехугольники, тетраэдры, гексаэдры) с возможностью расширения поддерживаемых типов сеток;
- реализованные численные алгоритмы должны включать в себя: КГид модель течения вязкой несжимаемой жидкости, модель

теплопроводности, модель конвекции-диффузии эволюции потока, электростатическую модель. Также должна быть предусмотрена возможность расширения набора моделей;

– возможность динамического конструирования составной модели из списка реализованных посредством конфигурационного файла.

Кроме того, желательна известная степень независимости моделей в том смысле, что моделирование течения применимо не только для текущих задач фильтрации, но и в рамках иных моделей, нуждающихся в определении распределений скорости и давления.

### **2.2.1 Параллельные алгоритмы решения задач**

Разработанные численные алгоритмы были адаптированы к параллельным вычислениям. В качестве основы для распараллеливания использовался принцип геометрического параллелизма.

Рассмотрим сначала параллельные численные аналоги макромоделей. Они реализуются разбиением неструктурированной сетки на компактные домены примерно одинаковой мощности (см. рис. 2.2). В качестве алгоритма разбиения возможно было использовать метод инерциальной бисекции с единичными весами в центрах ячеек [82], позволяющий добиться хорошей степени сбалансированности и компактности при разделении областей простой формы. Однако существуют и другие широко применяемые библиотечные решения, такие как пакеты Metis, ParMetis и Jostle, в совокупности, рассмотренные в работе [83] и обеспечивающие декомпозицию расчетной сетки произвольного типа и структуры.

При выборе конкретного метода декомпозиции было учтено, что при увеличении числа разбиений существенно падает эффективность параллельного алгоритма декомпозиции, особенно в случае разбиения неструктурированных сеток. Это в первую очередь связано с существенными накладными расходами при согласовании границ доменов разными вычислителями. Кроме того, численные схемы на неструктурированных

сетках приводят к дополнительным сложностям при формировании блоков памяти для проведения расчетов на графическом ускорителе. В результате декомпозиция расчетной области и последующий расчет на основе макромодели решено было реализовать на базе центральных процессоров. Дополнительной оптимизацией в этом случае было применение легких процессов (поток), позволяющих использовать возможности работы с общей памятью, при расчетах в рамках одного вычислительного узла.

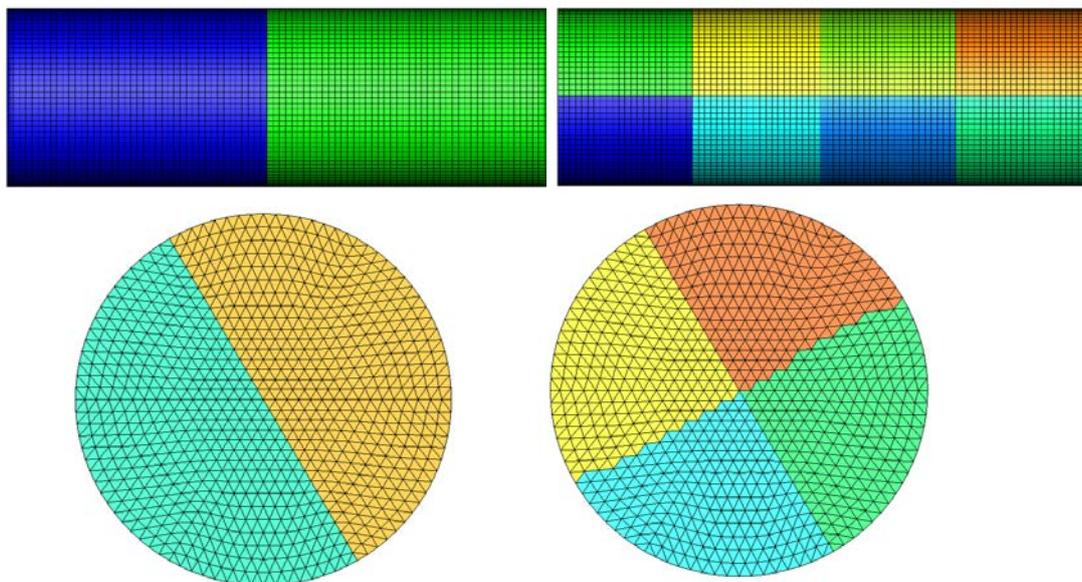


Рисунок 2.2 – Пример деления сеток на домены

Для микромоделей параллельные алгоритмы расчета являются комбинацией геометрического и функционального параллелизма. Для этого используется методика обрезания потенциалов взаимодействия частиц на некотором радиусе, вследствие чего вся расчетная область распадается на вычислительные домены. В итоге, все частицы попадают в тот или иной пространственно локализованный домен. Между доменами на шаге интегрирования по времени связей не возникает. Этот факт является основанием для прикрепления каждого домена или группы доменов к отдельному вычислительному узлу МВС. Таким образом, эффективно реализуется концепция геометрического параллелизма.

Внутри каждого домена производится расчет сил взаимодействия частиц между собой. Здесь ускорение расчетов достигается за счет использования параллельных потоков. Каждый параллельный поток рассчитывает силы взаимодействия между конкретными частицами. Поскольку общее количество частиц в данных задачах велико (в совокупности оно может составлять десятки и сотни миллионов), то в каждом домене их тоже достаточно много. Поэтому распараллеливание по потокам приводит к стабильно высокой эффективности вычислений. Это обстоятельство также позволяет использовать для численного расчета уравнений динамики частиц графические ускорители. Подробно параллельные алгоритмы такого типа обсуждаются в работе [84].

### **2.2.2 Расчетные коды**

При выборе языка программирования для разработки комплекта расчетных программ, адаптированных к использованию на гибридных суперкомпьютерных платформах существуют несколько критериев, в том числе: необходимость разработки эффективного, расширяемого и поддерживаемого кода на одном из общеупотребительных языков высокого уровня, используемых в научной и инженерной среде (например, C, C++, Fortran, Pascal и др.); наличие в языке поддержки стандартов межпроцессорного взаимодействия (например, MPI, OpenMP, CUDA и др.); совместимость с используемыми системами управления задачами пользователей.

Оптимальным решением для реализации расчетной части стало использование языка программирования C++, который является стандартом при разработке высокопроизводительных расчетных кодов, наряду с C и Fortran. Одним из основных преимуществ C++ является широкий набор поддерживаемых парадигм, в том числе объектно-ориентированное и шаблонное программирование, позволяющие выработать расширяемую

архитектуру вычислительного комплекса, простого в последующей поддержке [85].

Язык программирования C++ в целях организации параллельных приложений для межпроцессорного взаимодействия использует стандарт MPI, реализуемый конкретными подключаемыми библиотеками, в том числе OpenMPI [86]. Многопоточность достигается путем использования директив библиотеки OpenMP [87], включенной в большинство дистрибутивов операционной системы Linux. Организация вычислений на видеоускорителях осуществляется с помощью NVidia Cuda Toolkit, набора вспомогательных библиотек и обертки над компилятором GCC – утилиты nvcc [88].

Итоговая реализация программного комплекса состоит из набора библиотечных решений, схема которой изображена на рис. 2.3:

- библиотека представления конфигурации вычислительного ядра (класс GlobalConfig);
  - библиотеки представления матриц и векторов, а также операций над ними (пространство имен MathLib);
  - библиотеки представления отношений данных различной природы (пространство имен RelationLib);
  - библиотеки сеточного представления (пространство имен MeshLib);
  - библиотеки решателей, включающей в том числе представление данных (пространство имен SolverLib);
- и пакетной головной программы макромодели, динамически конфигурируемой при запуске с возможностью определения типа задания.

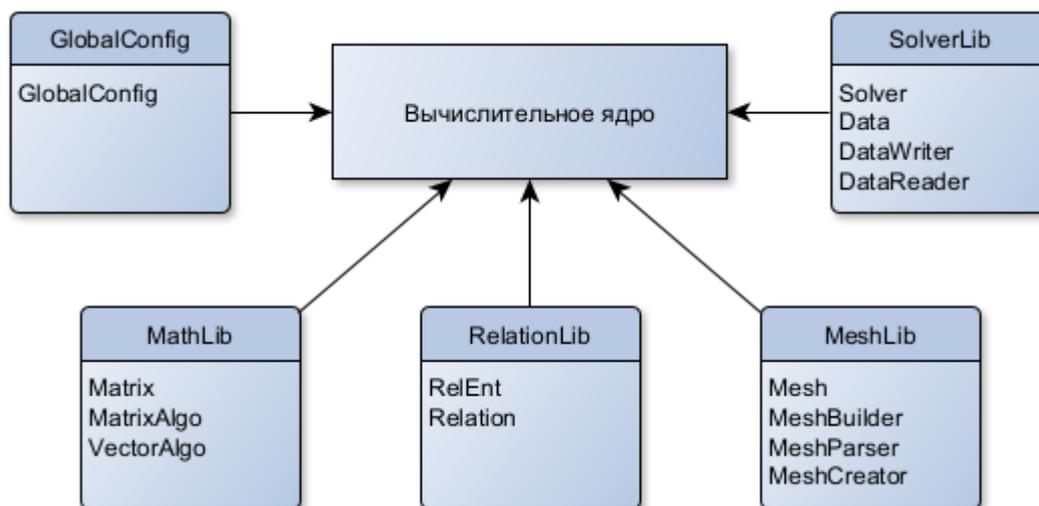


Рисунок 2.3 – Схема вычислительного ядра

### 2.2.2.1 Библиотека конфигурирования

Конфигурирование расчетного ядра обеспечивается с помощью статического класса `GlobalConfig`, реализующего классический шаблон «Одиночка». Применение данного подхода позволяет использовать единый объект в рамках вычислительного процесса.

В качестве формата файла конфигурации используется JavaScript Object Notation (JSON) представление [89], позволяющее описывать как элементарные типы данных (числа, строки, логические), так и комплексные (массивы, объекты). Формат JSON является понятным человеку, стандартизирован и имеет множество готовых решений для сериализации/десериализации для различных языков программирования (в таких языках, как Python и JavaScript поддерживается в рамках стандартной библиотеки).

Для обеспечения поддержки JSON в расчетном ядре используется библиотека `nlohmann::json` [90], обладающая высокой эффективностью и скоростью сериализации/десериализации, удобством использования (взаимодействие с объектом класса `nlohmann::json` подобно контейнерам `map` и `vector` стандартной библиотеки C++), а, кроме того, модульными тестами для каждого участка кода.

Текущая реализация класса `GlobalConfig` включает в себя, как приватный член класса, объект `plohmann::json`, не предоставляя прямого доступа. Получение данных конфигурации осуществляется посредством вызова шаблонного метода `getValue`. Чтение файла конфигурационного файла и формирование внутреннего представления осуществляется посредством метода `parseFile`.

### 2.2.2.2 Библиотека математических векторных и матричных операций

Представление таких величин, как координаты вершин сетки и ее векторные параметры (нормали, направленные площади и др.), списки и значения векторных переменных, в вычислительном ядре описывается с помощью классов пространства имен `MathLib`, структура которой изображена на рис. 2.4.

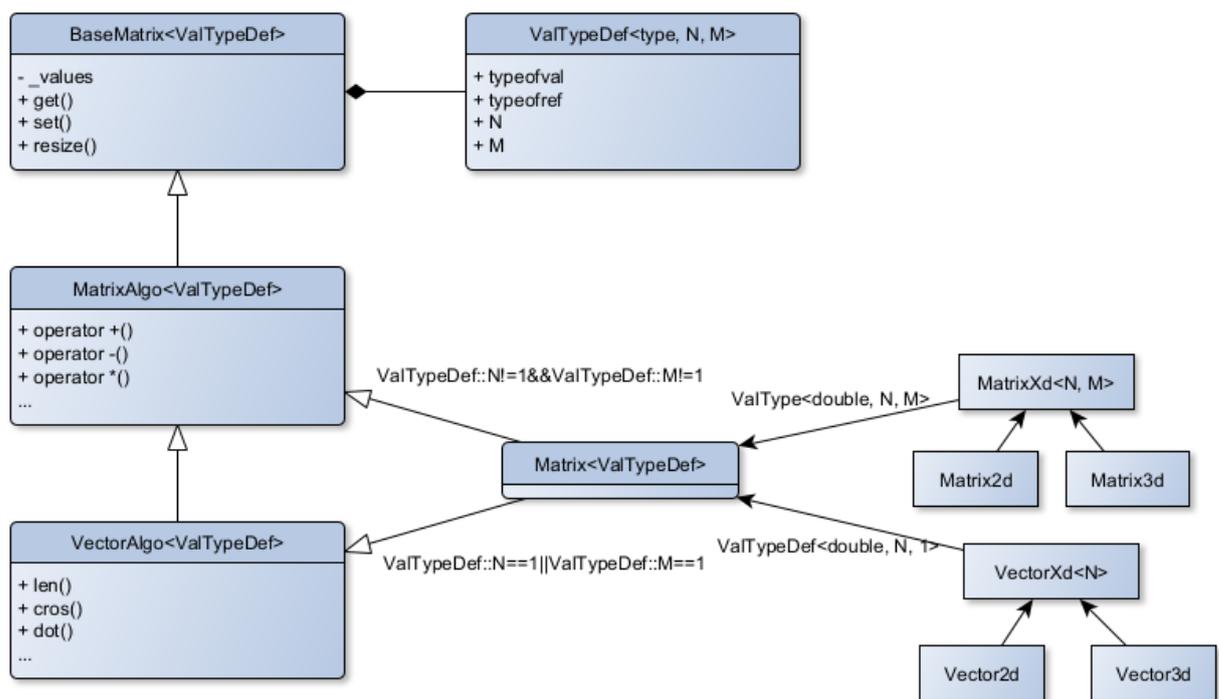


Рисунок 2.4 – Структура пространства имен `MathLib`

Повсеместное использование матриц и векторов вынуждает использовать шаблонный подход, позволяющий существенно ускорить, в сравнении с объектно-ориентированным, конструирование экземпляров и

вызов методов, поскольку уточнение полиморфизма происходит в момент компиляции, а не при исполнении. Более того, применение шаблонов позволяет определить ошибки, связанные с недопустимостью некоторых матричных и векторных операций (например, сложение матриц разной размерности или некорректное их умножение), на момент компиляции.

Основным базовым шаблонным классом является `BaseMatrix`, ожидающим в качестве параметра структуру `ValTypeDef`. Данная структура содержит тип хранимых значений `typeofval`, количество столбцов определяет член `N`, количество строк – `M`, тип массива содержится в `typeofref` и зависит от `N` и `M`. Если одно из них принимает нулевое значение, тогда `typeofref` представляется контейнером стандартной библиотеки `vector`, иначе `typeofref` описывает статический массив. В соответствии с переданным описанием `ValTypeDef` член `BaseMatrix` `_values` является либо `std::vector<ValTypeDef::typeofval>`, либо `typeofval[]`. Отметим, что двумерные матрицы представляются одномерными массивами либо векторами. Основной задачей базового класса является хранение и обеспечение доступа к данным, а также возможность изменения размера матрицы, при описании динамическим контейнером.

Класс `MatrixAlgo` наследует `BaseMatrix` и расширяет его общими матричными операциями, такими как сложение матриц, умножение матрицы на матрицу, на скаляр и другие. Класс `VectorAlgo` наследует `MatrixAlgo` такими методами, характерными для векторов, как скалярное и векторное умножения, нормировка вектора. Итоговый прикладной класс `Matrix` наследуется либо от `MatrixAlgo`, если значения `ValTypeDef::N` и `ValTypeDef::M` не равны единице, либо от `VectorAlgo`, в ином случае.

Для удобства использования библиотека содержит набор определений для часто применяемых типов матриц и векторов, например матрица  $3 \times 3$  содержащая значения двойной точности (`Matrix3d`) или двумерный вектор (`Vector2d`).

### 2.2.2.3 Библиотека отношений

Для описания сеток, сеточных элементов и разметок будем использовать подход неявных комплексов, описанных в работах [91, 92]. Для его реализации необходим механизм описания отношений между элементами различных типов и размерностей, например ячейка и узлы ее составляющие, грани и ячейки ее содержащие. Поскольку некоторые подобные отношений имеют статически определенное количество взаимоотношений (гексаэдральные ячейки состоят из восьми вершин, грани принадлежат двум ячейкам или являются граничными/межпроцессорными) и неопределенное (отношение разметка-грань, вершина-сеточный элемент) необходимо предусмотреть оба данных варианта. Схема пространства имен RelationLib изображена на рис. 2.5. Для хранения информации о взаимоотношениях используется вектор стандартной библиотеки `_rel` для типа `std::size_t`, содержащей индексы атрибутов или элементов.

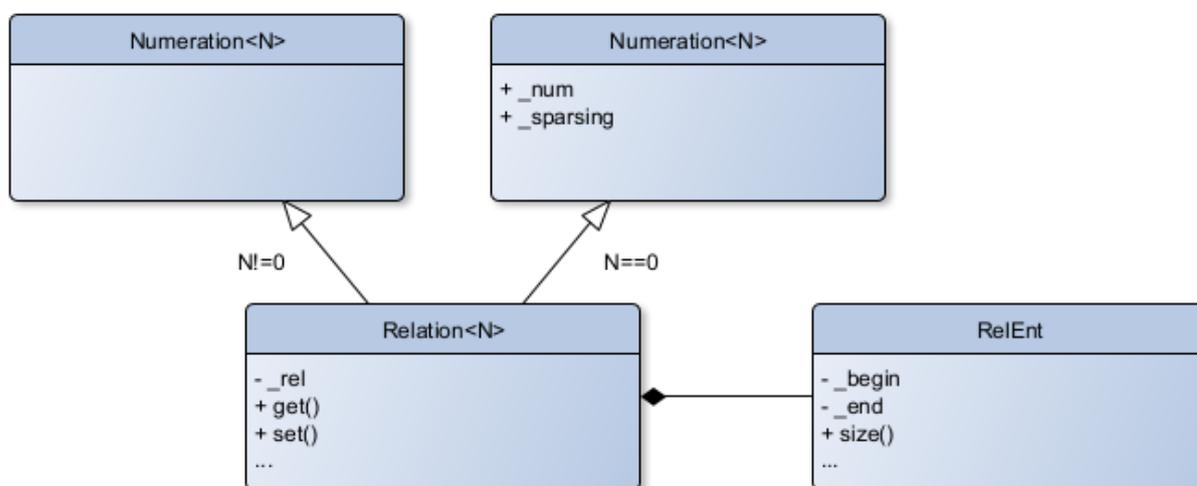


Рисунок 2.5 – Пространство имен RelationLib

Параметр шаблона  $N$  позволяет задать количество взаимоотношений в рамках одной записи. При  $N$  не равным нулю `Relation` наследует пустой класс `Numeration<N>`, а член `_rel` формируется как  $N * eICnt$ , где `eICnt` – количество элементов рассматриваемого типа. Отношение `RelEnt` в данном случае

формируется при вызове метода `get` с параметром `id`, с определением `_begin` как  $N*id$ , `_end` –  $N*(id+1)$ , где `id` – индекс элемента.

При параметре шаблона `N` равным нулю `Relation` может содержать отношения различного размера и наследует классу `Numeration` с членами `_num` и `_sparsing`. Член класса `_num` содержит индексы массива `_rel`, позволяющие формировать `RelEnt` с определением `_begin` как `_rel[_num[id]]`, а `_end` как `_rel[_num[id+1]]`. Целочисленный член класса `_sparsing` предназначен для установления разреженности массива `_rel`, при котором для `rel` устанавливается размер `_sparsing*elCnt`, а массив `_num` содержит величины  $i*_sparsing$ , для  $i = 0, \dots, elCnt+1$ . Установление ненулевой разреженности позволяет выделить необходимую память и сформировать массив `_num` с константным шагом индексов, что может существенно снизить нагрузку при добавлении элементов в массив отношений, как в части выделения памяти, так и в части переиндексации в `_num`. При этом возможность увеличения количества отношений сохраняется.

#### 2.2.2.4 Библиотека сеточного представления

Внутреннее представление неструктурированных разностных сеток, содержащее геометрические (координаты вершин), топологические (отношения типа сеточный элемент-вершина, грань-элемент) описания и атрибутирование (граничные поверхности, разметки объемов) сеток описывается с помощью базового виртуального класса `Mesh`, который в последствии уточняется до конкретного типа сеток. Конструирование класса осуществляется посредством `MeshCreator` в соответствии со схемой, изображенной на рис. 2.6. Посредством шаблонного параметра `MeshDim` определяется размерность разностной сетки – двумерная или трехмерная.

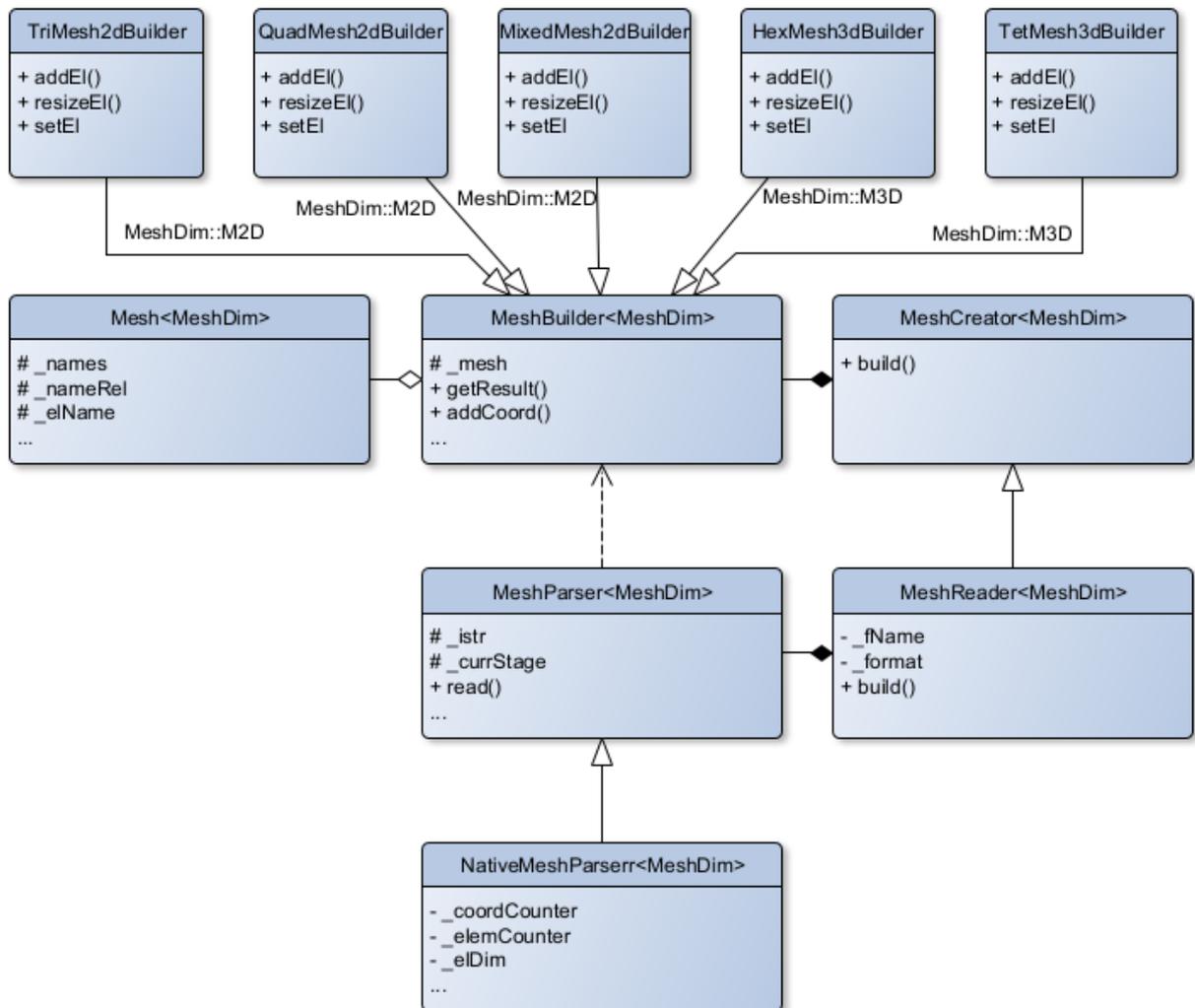


Рисунок 2.6 – Пространство имен MeshLib. Конструктор сетки

Процесс формирования сеточного представления реализуется посредством паттерна «Строитель». В качестве класса-супервизора выступает MeshCreator, наследуемый конкретными методами конструирования сеток. На текущий момент реализована возможность чтения сеток из файлов внутреннего формата посредством класса-реализации MeshReader, в рамках которого содержится объект класса MeshParser, выступающий интерфейсом для чтения и разбора файлов сеточного представления, реализацией которого является класс NativeMeshParser. Класс MeshBuilder, указатель на объект которого содержится в MeshCreator, наследуется классами-строителями различных типов сеток, включающих такие, как треугольные (TriMesh2dBuilder), четырехугольные (QuadMesh2dBuilder), гексаэдральные (HexMesh3dBuilder) и сетки других

типов. Представленная архитектура поддерживает расширяемость как списка конструируемых типов сеток, так и методов их формирования.

Подробнее рассмотрим структуру сеточного представления, реализованного в рамках вычислительного ядра, посредством схемы, изображенной на рис. 2.7.

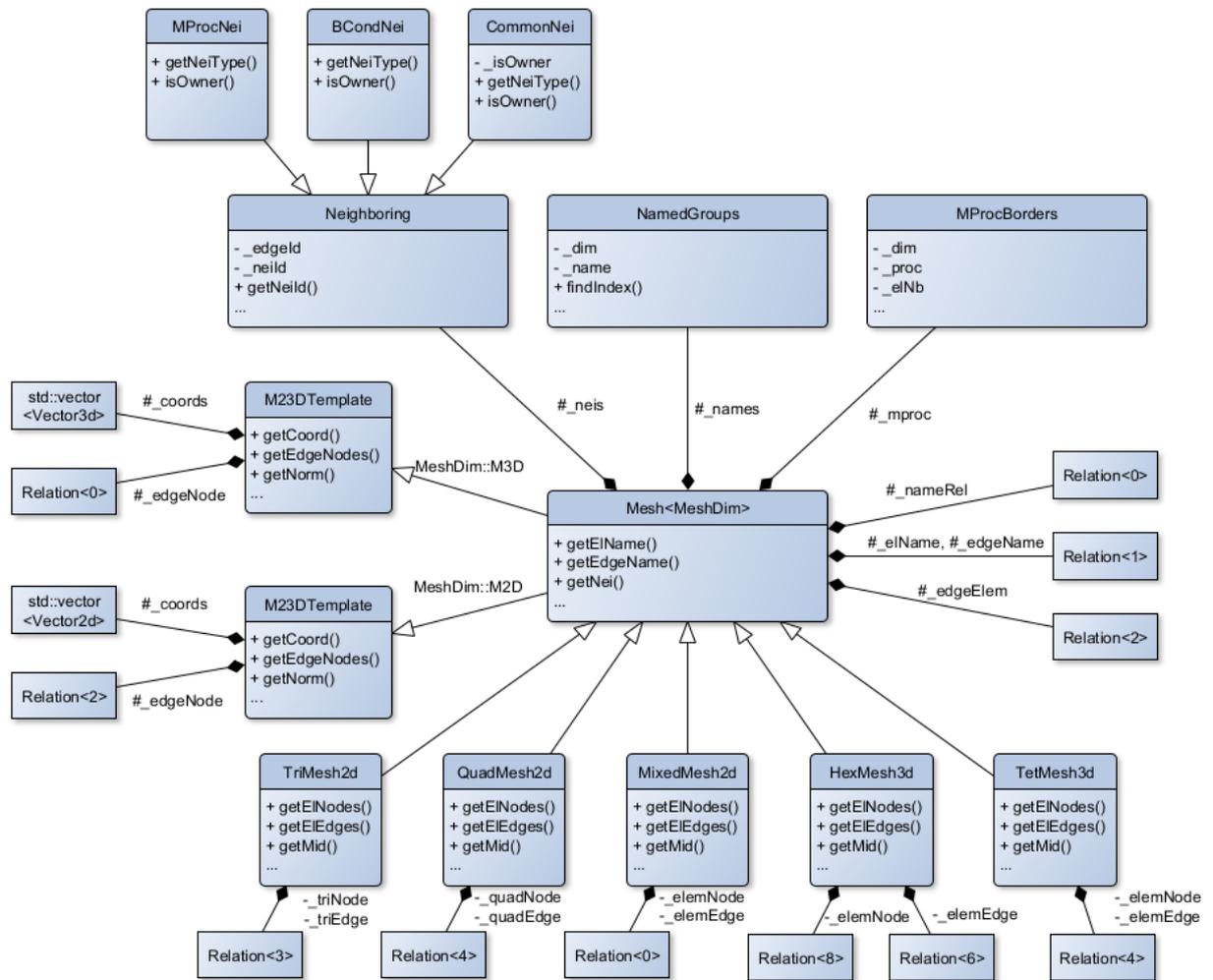


Рисунок 2.7 – Пространство имен MeshLib. Сеточное представление

Базовый шаблонный класс `Mesh` наследует `M23DTemplate` в реализации, соответствующей требуемой размерности сетки. `M23DTemplate` содержит список координат (`_coord`) и отношения грань-узлы (`_edgeNode`), отвечает за обеспечения доступа к данным членам класса, а также набор вспомогательных функций. При наследовании двухмерного варианта `_coords` содержит элементы типа `Vector2d`, а `_edgeNode` представляет собой объекты класса `Relation<2>`, поскольку грани в данном случае представляются

линиями. Трехмерный вариант представляет координаты как список элементов `Vector3d`, `_edgeNode` как `Relation<0>` - в данном случае гранями трехмерных сеточных элементов могут выступать треугольники, четырехугольники и иные плоские фигуры. Непосредственно класс `Mesh` содержит метаинформацию, включающую информацию о наименовании (разметке) элементов разной размерности, данные о межпроцессорных связях и типе соседства сеточных элементов (обычное, граничное, межпроцессорное).

Конкретные сеточные представления описываются классами: `TriMesh2d` – двумерная треугольная сетка, `QuadMesh2d` – двумерная четырехугольная сетка, `MixedMesh2d` – двумерная сетка, содержащая элементы различных типов, `HexMesh3d` – гексаэдральная трехмерная сетка, `TetMesh3d` – тетраэдральная трехмерная сетка. Указанные сущности наследуют базовый шаблонный класс `Mesh`, уточняя его размерность. Данные, содержащиеся в классах представляют из себя отношения элемент-узлы, данное отношение фактически описывает геометрию сеточного элемента, и элемент-грань, отражающее конкретное соседство.

Описанная архитектура допускает расширение набора поддерживаемых типов сеток, текущая же реализация направлена по большей части на конечно-объемные разностные схемы с конечным объемом – сеточным элементом.

### **2.2.2.5 Библиотека решателей**

Расчетный код, реализующий описанные в п. 2.1 разностные схемы, основан на классе `Solver`, схема которого представлена на рис. 2.8. Композиция моделей достигается применением паттерна «Декоратор», при котором ссылка на предыдущий в списке конфигурации решатель содержится в текущем. Основная информация о задаче, включающая расчетную сетку, рассчитываемые величины, сервисные данные и классы инициализации данных содержатся в первом объекте решателя. Информация

о граничных условиях содержится в каждом из потомков, реализующих соответствующую модель численной методики. Список соответствующих классов включает в себя следующие элементы:

- QHydroDSolver – КГид модель;
- HCSolver – модель теплопроводности;
- CDSolver – модель конвекции-диффузии;
- EMSolver – электростатическая модель.

Конфигурирование итоговой композиции решателей осуществляется вызовом метода `configure`, а последующая инициализация вызовом метода `initialize`. В рамках текущей версии расчетного кода для инициализации данных предусмотрено две возможности – инициализация константным значением (`ConstInitializer`) и инициализация зависимостью от координат (`CoordInitializer`). Процесс расчета формируется вызовом метода `iterate` в цикле, с остановом по условию – достижение максимального количества шагов или требуемого расчетного времени.

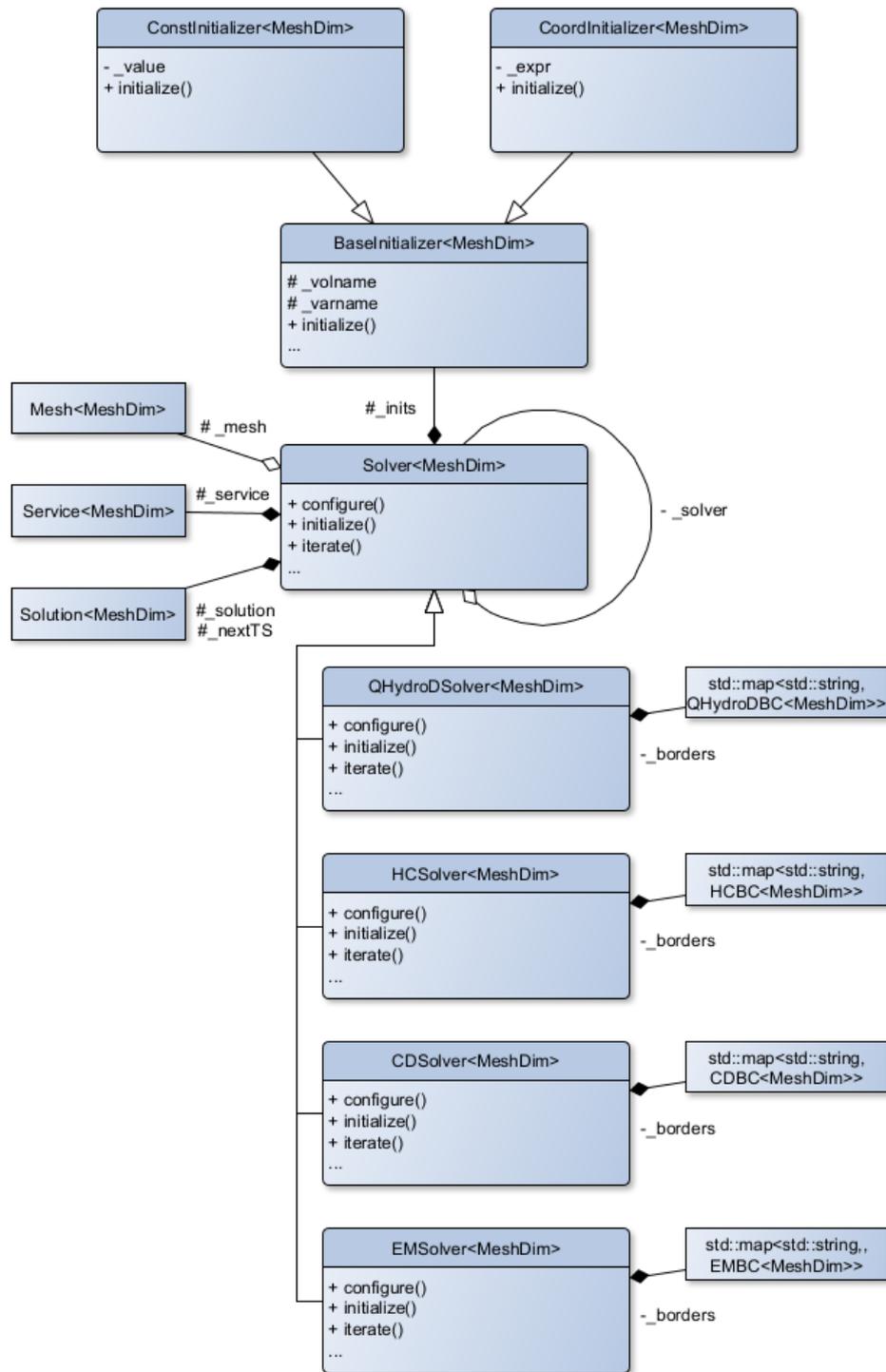


Рисунок 2.8 – Пространство имен Solvers. Класс вычислителей

Рассчитываемые величины содержатся в объекте класса Solution, наследующего класс Data, схема которого изображена на рис. 2.9. В классе Data, данные разделяются на типы – векторные и скалярные данные, которым отвечают `_scalarVars` и `_vectorVars` соответственно. Класс Service

предназначен для хранения дополнительных сервисных данных, таких как объемы сеточных элементов, площади граней, нормали и другие.

Чтение и запись рассчитываемых величин осуществляется посредством абстрактных классов `SolutionReader` и `SolutionWriter`, потомки которых реализуют данный функционал в требуемом формате. В текущей версии вычислительного ядра реализована поддержка форматов библиотеки научной визуализации VTK [93] посредством классов `SolutionVTKReader` и `SolutionVTKWriter`.

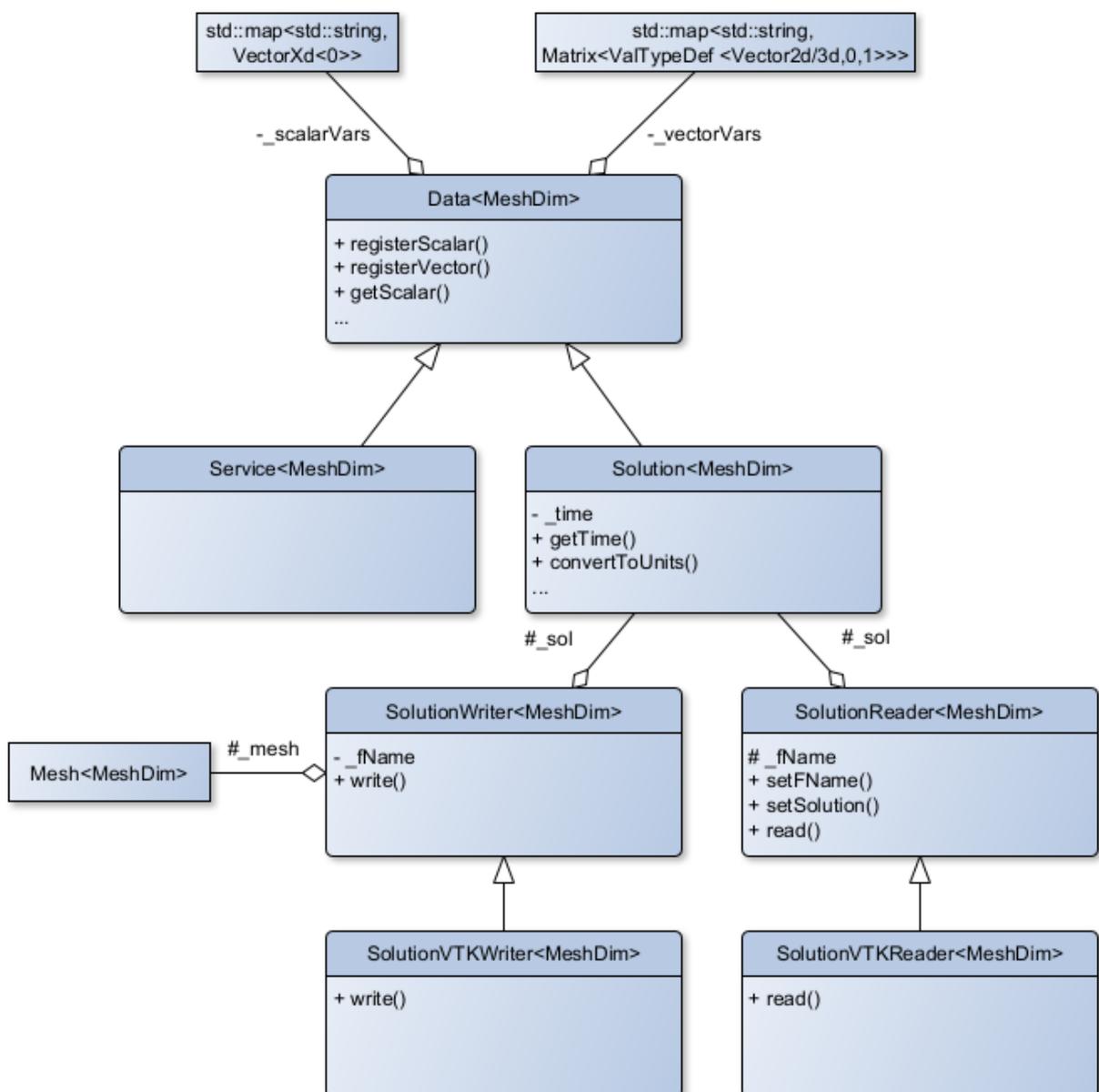


Рисунок 2.9 – Пространство имен Solvers. Классы данных

Схема классов, описывающих граничные условия, представлена на рис. 2.10. Граничные условия разделены на группы, соответствующие классам вычислителей:

- QHydroDBC – граничные условия КГид модели, включающие граничные условия по скорости (QHydroDVelocity), по давлению (QHydroDPressure) и условие прилипания (QHydroDWall);
- HCBC – граничные условия теплопроводности, позволяющие задать температуру (HCBCTemperature), тепловой поток (HCBCFlow) и теплообмен с окружающей средой (HCBCConvection);
- CDBC – граничные условия модели конвекции-диффузии, включающие возможность задания концентрации (CDBCConcentration), потока (CDBCFlow) и адсорбции (CDBCAdsorbtion);
- EMBC – граничные условия электростатической модели, включающие задание электронейтральных границ (EMBCNeutral) и электрического потенциала (EMBCCurr).

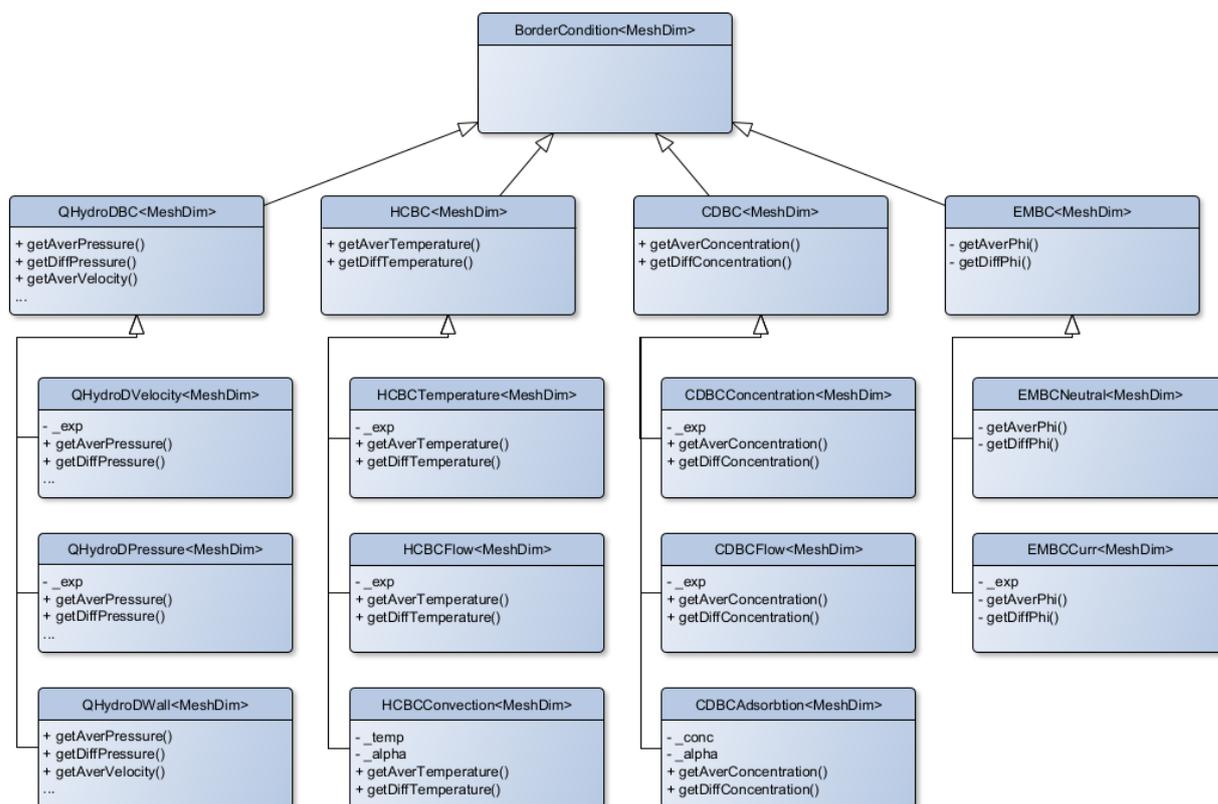


Рисунок 2.10 – Пространство имен Solver. Классы граничных условий

### 2.3 Выводы по Главе 2

Во второй главе представлены численные алгоритмы, основанные на методе конечных объемов, приведенных в первой главе моделей. Предложена архитектура расширяемой программной реализации данных алгоритмов, позволяющая использование неструктурированных сеток различных типов.

Результаты работ [24 – 36] в части МСС получены с помощью вычислительных алгоритмов и программного ядра, описанного в данной главе.

## **Глава 3. Цифровая платформа**

### **3.1 Общие положения и предпосылки**

С ростом производительности центральных процессоров, а также применением видеокарт в качестве базовых вычислительных устройств стало возможным решение многих сложных научных и инженерных задач на персональных компьютерах и небольших системах коллективного пользования. Одновременно с этим шло развитие статистических и прямых методов математического моделирования, которые смыкаются в настоящее время с технологиями нейросетей и искусственного интеллекта. В результате объединение традиционных и новых подходов в математическом и компьютерном моделировании стало повседневной практикой. Обычная и высокопроизводительная техника становится все более доступной во многих исследовательских организациях и на производстве. В связи с этим возникает задача создания аппаратно-программных платформ для решения большого круга научно-технических проблем. Основной целью подобных систем является существенное упрощение и автоматизация большинства этапов проведения вычислительного эксперимента, включающих подготовку исходных данных, управление и слежение за расчетным процессом, анализ результатов расчета. Реализация современных цифровых платформ, безусловно, должна учитывать тенденции развития вычислительной техники, в том числе высокопроизводительной.

Основу вычислительных кластеров и суперкомпьютеров сегодня составляют гибридные системы, построенные в рамках модели распределенных вычислений, в которой множество гибридных вычислительных узлов (имеющих в своем составе центральные процессоры и альтернативные устройства обработки данных, например графические ускорители) объединяется высокоскоростной сетью [94]. Примером таких систем служат гибридные вычислительные кластеры ИПМ им. М.В. Келдыша РАН – K100 и K60 [95]. Именно такие системы в конечном итоге

способны развиваться продолжительное время и адаптироваться к новым все возрастающим потребностям в вычислениях, допуская увеличение вычислительного поля, как посредством увеличения количества узлов, так и модернизации уже существующих.

В части системного программного обеспечения (ПО) современные гибридные кластеры и суперкомпьютеры управляются UNIX-подобными операционными системами, обеспечивающими многозадачный и многопользовательский режим работы, в том числе в рамках удаленного доступа, при соблюдении всех необходимых стандартов безопасности. При этом для эффективной организации параллельных вычислений используются различные интерфейсы и средства параллельного и гибридного программирования (MPI, OpenMP, CUDA Toolkit и др.), а также системы управления заданиями пользователей. Основная цель последних – планирование вычислений, обеспечение доступа к вычислительным узлам кластера, организация очереди и управление расчетными заданиями пользователей (от планового выделения ресурсов и запуска заданий, до слежения за состоянием конкретного задания и гарантированного сохранения результатов проводимых в его рамках расчетов). Примером такой системы в России является СУППЗ [96], разработанная сотрудниками ИПМ им. М.В. Келдыша РАН и ФГУП "НИИ "Квант". Широко известны и распространены также альтернативные программы организации очереди, такие как Slurm [97].

Стоит отметить также современный подход, основанный на технологиях облачных вычислений, представляющий альтернативу классическим системам совместного использования вычислительных мощностей, основным принципом которого является предоставление ресурсов по требованию посредством слоя виртуализации. Пользователю предлагается либо доступ к виртуальной машине, контейнеру Docker [98], системе контейнеров Kubernetes [99] посредством стандартного SSH-соединения, либо специализированный сетевой программный интерфейс приложения (API). Сегодня на рынке существует множество решений,

предоставляющих услуги облачных вычислений, таких как Microsoft Azure [100], Amazon Web Services [101], отечественным аналогом которых может выступать Yandex Cloud [102]. Каждый из указанных сервисов предлагает собственную систему вызовов для выделения ресурсов, запуска и слежения за вычислительным процессом.

Помимо системного ПО требуется поддержка в виде библиотек и прикладных параллельных программ, позволяющих решать задачи из разных областей вычислительной и прикладной математики. Немаловажную роль при этом играют средства и технологии доступа к вычислительным ресурсам, средства подготовки расчетных данных и средства постобработки и визуализации получаемых результатов.

### **3.2 Архитектура цифровой платформы**

Развитие и повсеместное применение веб-технологий привлекло внимание автора при исследовании вопроса создания предлагаемой цифровой платформы для проведения расчетов, нацеленных на суперкомпьютерное моделирование сложных физических процессов [103, 104]. Мотивацией к использованию веб-технологий стало то обстоятельство, что при создании полноценного графического интерфейса, ориентированного на пользовательский компьютер, существует проблема производительности последнего, а также качества предустановленного на нем программного обеспечения. Применение веб-технологий исключает данные вопросы – для использования приложения достаточно наличия современного веб-браузера. Вся тяжесть предобработки и графического представления данных ложится на сервер, в качестве которого может выступать суперкомпьютер, на базе которого были получены основные результаты моделирования.

Помимо сказанного существуют дополнительные преимущества клиент-серверного подхода, которые включают:

- обновление программного обеспечения не зависит от конечного пользователя, так как данная проблема находится в области ответственности

администратора сервера;

- возможность совместной работы пользователей над одним расчетным проектом (многопользовательский режим);

- независимость прикладной программной среды в целом от состояния персонального компьютера пользователя, что особенно актуально при мониторинге прохождения задач на удаленных вычислителях и своевременной обработке расчетных данных;

- унификация доступа к вычислительным ресурсам различного типа в рамках единого подхода.

Основные существующие реализации подобных платформ можно разделить на две категории. С одной стороны, это веб-лабораторий, такие как [105, 106], позволяющие проводить суперкомпьютерное моделирование в рамках отзывчивого пользовательского веб-интерфейса. При этом, однако, в подобные системы направлены на решение конкретного класса задач. Также подобные системы не предусматривают свободного динамического расширения интегрированных вычислительных устройств. Другой тип систем [107, 108], предлагает удобное администрирование вычислительных ресурсов и динамическое добавление прикладных программных кодов в систему. Однако в этом случае конфигурирование прикладных программ осуществляется посредством стандартных строковых/числовых/флагов полей. Кроме того, доступ к выходным данным прикладных приложений реализуется посредством стандартного файлового интерфейса, обеспечивающего простейший просмотр или загрузку для локальной обработки.

В настоящей работе при создании прототипа цифровой платформы были учтены все вышеуказанные обстоятельства и приняты решения, направленные на создание своего рода вычислительной среды, в которой будет возможно проведение полномасштабных компьютерных и суперкомпьютерных экспериментов по выбранной прикладной тематике на доступных гибридных вычислительных кластерах в режиме удаленного

доступа, для чего предпринята попытка объединения представленных подходов. Ее структурная схема изображена на рис. 18 и отражает основные требования, накладываемые на разрабатываемую систему:

- многопользовательский доступ по протоколу HTTP/HTTPS;
- взаимодействие с удаленными вычислителями по SSH с возможностью расширения поддерживаемых протоколов (например, специализированного API конкретных решений для облачных вычислений);
- авторизованный доступ пользователей к системе, разделение пользователей по ролям (уровням доступа);
- обеспечение клиентов графическим интерфейсом для взаимодействия с удаленными вычислителями;
- обеспечение информационной поддержки о состоянии вычислителя, прикладных программных кодах, проведенных расчетах;
- наличие расширяемой базы данных.

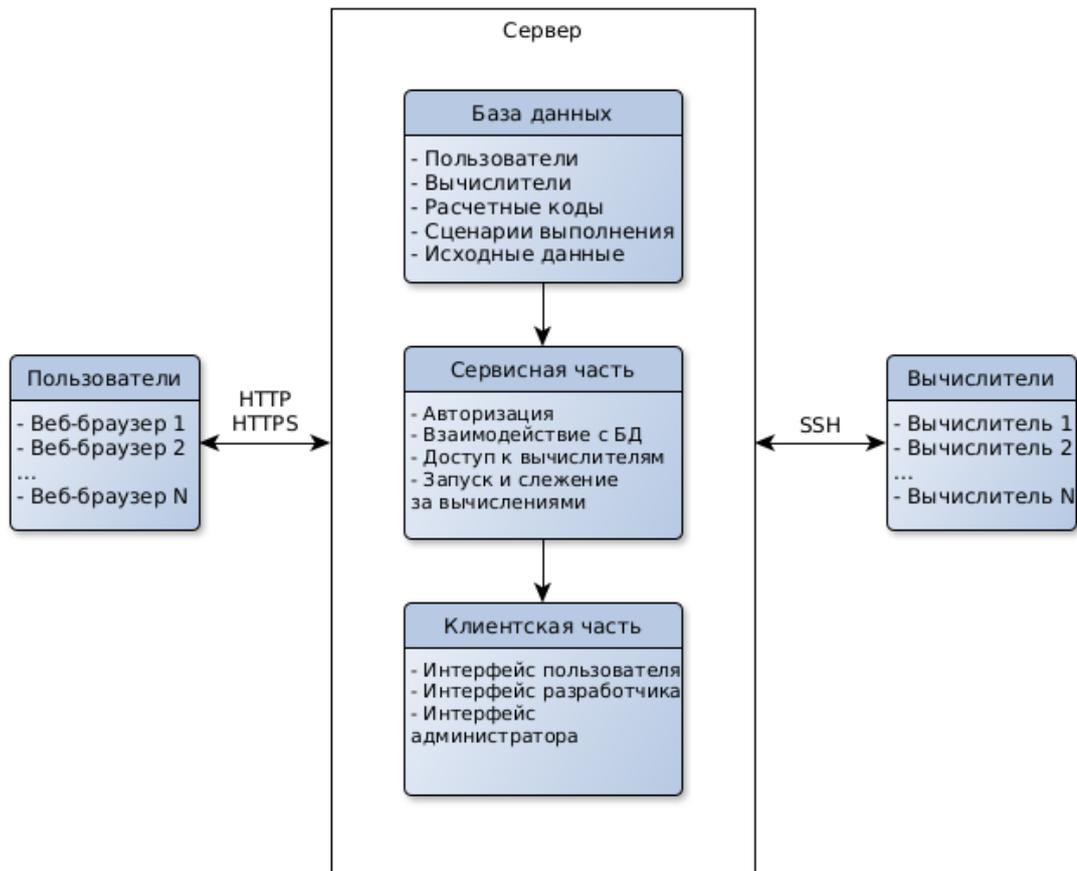


Рисунок 3.1 – Структурная схема цифровой платформы

Рассмотрим подробнее особенности и возможности системы, реализуемые выполнением основных требований.

В части безопасности, основывающейся на использовании шифрованного протокола HTTPS, веб-система позволяет исключить необходимость прямого доступа конечного пользователя к вычислительным ресурсам. Этот факт позволяет применить организацию сетевой инфраструктуры, в рамках которой суперкомпьютер или кластер располагается во внутренней сети, содержащей также веб-сервер платформы. При использовании подобного подхода для организации удаленного доступа достаточно разрешить использование во внешней сети порта, слушаемого веб-сервером. Поскольку цифровая платформа обрабатывает известный ограниченный набор запросов и сценариев взаимодействия с вычислительным ресурсом, существенно сокращается количество возможных направлений атаки.

Разделение ответственности пользователей поддерживается посредством следующих основных ролей: обычный пользователь (расчетчик), разработчик программного обеспечения и администратор системы. В зависимости от типа аккаунта цифровая платформа после авторизации предлагает модификацию графического веб-интерфейса, посредством которого, реализуется соответствующий роли функционал. Таким образом, расчетчику предоставлены возможности проведения вычислительных экспериментов, разработчику – регистрация в системе новых прикладных приложений, администратору – обслуживание системы и вычислительных ресурсов, администрирование базы данных.

Классический процесс проведения комплексного вычислительного эксперимента заключается в ручной подготовке начальных данных локально или удаленно с помощью SSH-клиента в терминале. При локальной подготовке полученные файлы необходимо поместить на удаленный вычислительный ресурс. После этого, посредством выполнения консольных команд, производится запуск задания в соответствии с подходом,

определенным на вычислителе (СУППЗ, slurm и т.д.), после завершения полученные данные необходимо подвергнуть тому или иному анализу. Использование цифровой платформы расчетчиком для проведения вычислительного эксперимента предполагает подготовку исходных данных, запуск, управление и слежение за прохождением вычислительного задания и анализ полученных результатов посредством графического пользовательского веб-интерфейса, без необходимости использования терминала и знания особенностей системы управления заданиями. Более того, набор приложений выполняющих конкретную задачу (примером может служить процесс генерации сетки с последующим ее разбиением) в системе предлагается объединять посредством сценариев исполнения – специальных текстовых файлов, описывающих процедуру и порядок запуска. При этом, все проведенные расчеты требуют наличия метаинформации, включающей название, текстовое описание, указание на используемый вычислительный ресурс, пути расположения файлов, сохраняемой в базе данных и доступных расчетчику как во время, так и после проведения расчета.

Разработчикам прикладных приложений применение предложенной клиент-серверной архитектуры позволяет проводить совместную разработку, ставшую практически стандартным подходом при разработке комплексных приложений. Дополнительным преимуществом является возможность обновления приложений для всех пользователей системы.

### **3.3 Прототип цифровой платформы**

В рамках диссертационной работы был получен рабочий прототип цифровой платформы, отражающий основные идеи и подходы, обсуждаемые в данной главе.

Полученное приложение представляет собой сервер, при обращении на корневой адрес которого клиент получает набор html-разметки, css-стилей и js-скриптов, формирующих в веб-браузере графический интерфейс. В результате пользовательского взаимодействия с формами и элементами

управления веб-браузер формирует запросы (как стандартные HTTP/HTTPS, так и через WS/WSS), отправляемые на сервер. При обработке запросов веб-сервер обращается либо к базе данных, хранящей требуемую информацию, либо открывает ssh-соединение на удаленный вычислительный ресурс и выполняет соответствующие запросу действия. После обработки запроса сервер направляет запрошенные данные веб-браузеру пользователя.

### **3.3.1 Технологический стек**

Первая разработка подобной веб-лаборатории силами нашего отдела [109 – 111] изначально основывалась на Django [112] – библиотеки для языка программирования Python, содержащей все необходимое для создания клиент-серверного решения, в том числе механизм объектно-реляционного отображения (ORM) для взаимодействия с базами данных, генератор веб-страниц из html-шаблонов, системы маршрутизации и авторизации. В процессе разработки выяснилось, что использования подхода шаблонных страниц недостаточно для создания интерактивного пользовательского интерфейса, обеспечивающего необходимые для проведения вычислительного возможности. Для поддержки требуемого функционала приходилось использовать JavaScript кода, исполняемый непосредственно в веб-браузере. В итоге от использования шаблонов было решено отказаться в пользу разработки одностраничного приложения (SPA) в реактивном стиле, для чего была применена библиотека Vue.js [113].

В данной разработке было решено исключить также и Django, поскольку большая часть его функционала, заключающаяся в шаблонизаторе, перестала использоваться.

Программной основой для серверной части системы был использован Node.js [114], являющийся расширением JavaScript. Обычно язык программирования JavaScript используется на стороне веб-клиента и поддерживается во всех современных браузерах. Однако он имеет существенные ограничения по функциональности, в том числе

невозможность взаимодействия с устройствами ввода-вывода и системными потоками напрямую. Для расширения JavaScript до языка общего назначения в целях создания серверного решения и используется Node.js.

Применение Node.js позволило, с одной стороны, добиться однородности кодовой базы между клиентской и серверной частью, а с другой – использовать высокую степень асинхронности JavaScript на сервере. Это особенно полезно для целей слежения за прохождением задач на удаленных вычислителях, выполнении длительных задач и масштабирования системы.

Для обеспечения маршрутизации, а также использования плагинов безопасности был использован фреймворк Express.js [115], авторизация реализована посредством библиотеки passport.js [116]. Взаимодействие с базой данных осуществлялось с помощью библиотеки ORM Sequelize [117]. В качестве базы данных на стадии разработки использовалась SQLite [118].

Поскольку создаваемая цифровая платформа является закрытой для общего доступа, было решено организовать клиент-серверное взаимодействие на основе веб-сокетов [119]. Веб-сокеты представляют из себя надстройку над протоколом HTTP. В отличие от исходного протокола, они поддерживают соединение после начального клиентского запроса о подключении. При успешном установлении такого клиент и сервер могут произвольно обмениваться сообщениями. Сообщения, отправляемые посредством веб-сокета, включают в себя исключительно тело запроса, а HTTP-заголовок повторно не отправляется, что позволяет существенно оптимизировать клиент-серверное общение. Для использования веб-сокетов на сервере применялась библиотека ws [120]. Обеспечение HTTPS-протокола достигалось посредством обратного прокси сервера nginx [121].

Клиентская часть была реализована с применением реактивной парадигмы, организованной с помощью использования фреймворка Vue.js и библиотеки компонентов Quasar [122]. Для SPA-маршрутизации применялось естественное в данном случае решение Vue-router [123], для

организации централизованного хранилища данных использовалась Vuex [124] (хранение информации о пользователе, открытого веб-сокета, уведомлений). В отношении компонента анализа результатов расчетов использован ParaViewWeb [125] с встраиванием клиентской части в реализации Visualizer.js [126]. Данное решение обеспечивает поддержку форматов VTK, возможность интерактивной серверной или клиентской отрисовки в зависимости от аппаратного обеспечения пользователя. Кроме того, ParaViewWeb поддерживает большое количество фильтров данных, сопоставимое со стандартным «толстым» приложением ParaView [127]. Отдельным преисуществом является возможность реализации собственного клиента на стандартном стеке цифровой платформы, отказавшись от встраивания Visualizer.js.

### 3.3.2 Структура базы данных

База данных прототипа содержала минимально необходимый для организации предложенного механизма работы цифровой платформы набор таблиц, схема взаимосвязей между которыми представлена на рис. 3.2.

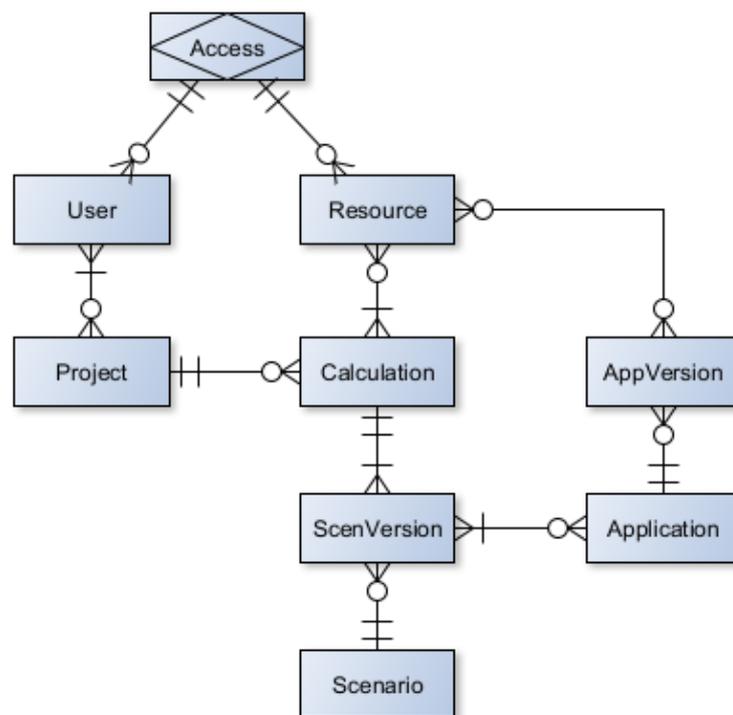


Рисунок 3.2 – Схема связей между сущностями базы данных

Для поддержки возможности динамической регистрации удаленного вычислительного ресурса необходима информация об адресе вычислительного ресурса, порте SSH-сервера, системе прохождения задач, возможностях задания переменных окружения. Эта информация размещается в базе данных цифровой платформы в таблице «Resource». Для осуществления SSH-доступа к удаленному вычислителю и разделения на нем пользователей, используется таблица «Access», содержащая ссылки на записи пользователя системы и вычислителя, а также дополнительные переменные окружения и указание на используемые пользователем версии приложений. Подключение к удаленному вычислителю производится с использованием RSA-ключа веб-системы.

Описание приложений, кроме информационного, включает в себя также задание шаблонного метафайла в формате `yaml`, содержащего инструкции сборки, запуска, описание входных и выходных параметров, хранение которого производится в таблице «Application». Поддержка версионирования достигается с помощью таблицы «AppVersion», содержащей дополнительно информацию о пути к исходному коду и итоговый метафайл, используемый для генерации графического пользовательского интерфейса.

Сценарии (таблица «Scenario») используются для объединения нескольких приложений в единый цикл запуска с возможностью связывания входных и выходных данных приложений, для описания используется `yaml`-файл определенной структуры. Версионирование происходит при помощи таблицы «ScenVersion», итоговое метаописание которого используется при генерации графического пользовательского интерфейса.

Сущность проекта (таблица «Project») носит информационный характер и используется для объединения в единую смысловую конструкцию нескольких расчетов. Расчеты (таблица «Calculation») являются основным способом взаимодействия с удаленным вычислителем, при создании которого задаются сценарий и его версия. В результате система генерирует

графический интерфейс для ввода исходных данных, управления расчетными заданиями и доступа к выходным данным на удаленном вычислителе.

### 3.3.3 Веб-сервер

Взаимодействие с сервером цифровой платформы основано на использовании графических и иных элементов пользовательского интерфейса, в результате взаимодействия с которыми веб-браузер отправляет на сервер соответствующий запрос. Маршрутизация запросов осуществляется посредством унифицированных указателей ресурса (URL) и имеет следующий вид:

- GET /auth/ - проверка актуальности сессии текущего пользователя;
- POST /auth/login/ - авторизация пользователя;
- POST /auth/logout/ - завершение сессии текущего пользователя;
- POST /auth/register/ - регистрация нового пользователя;
- GET /files/ - получение файла, расположенного в рабочей директории веб-сервера;
- GET /files/ssh/ - получение файла с удаленного вычислительного ресурса;
- POST /files/tmp/ - отправка пользовательского файла на сервер во временное хранилище;
- GET /api/ - при обращении открывает основное веб-сокеты соединение, предназначенное для обеспечения взаимодействия между клиентом (веб-браузером) и сервером цифровой платформы;
- GET /terminal/ - при обращении открывает веб-сокеты соединение с веб-браузером пользователя, а также устанавливает SSH-соединение с требуемым удаленным вычислителем. После успешной установки соединений сервер связывает веб-сокеты с SSH-клиентом.

Как указано выше, основным каналом взаимодействия между веб-браузером пользователя и сервером цифровой платформы является веб-сокеты,

открываемый при обращении по URL-адресу /api/. Сообщения, отправляемые клиентом должны содержать обязательное поле «url» для вызова корректного обработчика запроса и опциональное поле «data», содержащее добавочную информацию. Поле «url» должно представлять собой строку, во многом соответствующую стандартному URL-адресу. Ответ сервера отправляется в сообщении, содержащем поля «url», «body» и числового кода «status».

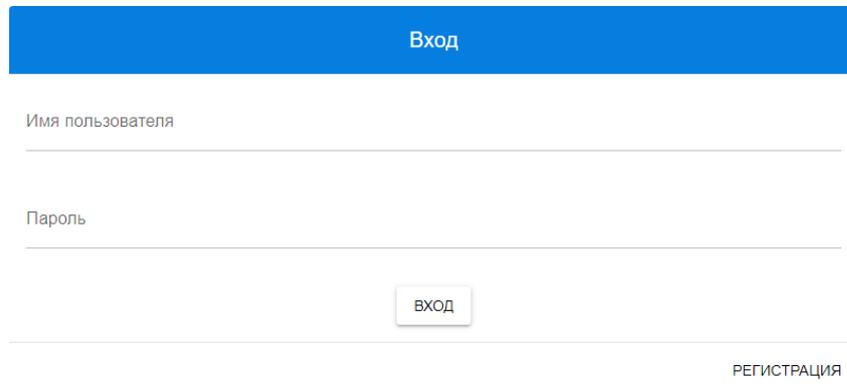
Необходимость наличия в ответе поля «url» объясняется особенностью работы веб-сокета, заключающейся в отсутствии классической схемы запрос клиента и последующий ответ сервера. При использовании соединения данного типа клиент может только лишь подписаться на сообщения, отправляемые сервером (установка обработчика onmessage). В рамках разработанного подхода единый обработчик входящих сообщений вызывает конкретные клиентские функции, установленные при отправке запроса на сервер, на основании поля «url» ответа. Различаются два типа устанавливаемых обработчиков – постоянные, при реагировании на сообщения о новом уведомлении или сообщении, подписке на изменение статуса расчетного задания, и одноразовые, используемые при запросе информации из базы данных, списка файлов на удаленном вычислителе.

Отметим также, что масштабирование веб-сервера производится посредством создания подпроцессов средствами Node.js. При этом главный процесс обеспечивает задачи слежения за расчетными заданиями и состоянием вычислительных ресурсов. Процессы-потомки используются для обработки пользовательских запросов и обслуживания установленных веб-сокет соединений.

### **3.3.4 Пользовательский веб-интерфейс**

Для обзора клиентской части разработанного прототипа цифровой платформы рассмотрим типовые сценарии ее использования, в том числе регистрацию вычислительных ресурсов, приложений, сценариев и проведения вычислительного эксперимента.

Доступ к цифровой платформе происходит по обращению пользователя через браузер его ПК по корневому URL-адресу. Установленное соединение приведет пользователя к форме авторизации, изображенной на рис. 3.3.



Вход

Имя пользователя

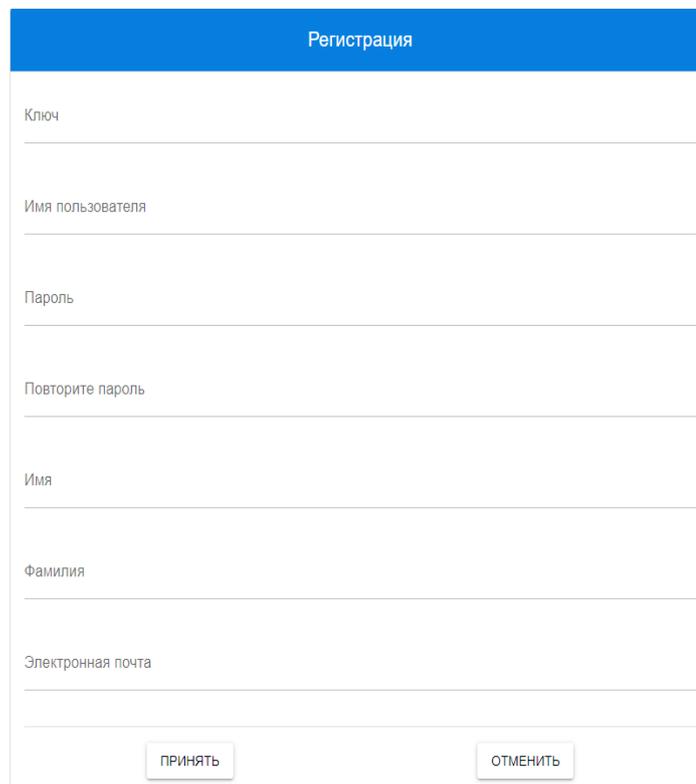
Пароль

ВХОД

РЕГИСТРАЦИЯ

Рисунок 3.3 – Форма авторизации

При отсутствии пользователя в системе предполагается его обращение к администратору системы с целью получения ключа доступа, генерируемого системой автоматически и определяющем тип аккаунта. При наличии ключа пользователь получает возможность пройти регистрацию, посредством соответствующей формы (см. рис. 3.4).



Регистрация

Ключ

Имя пользователя

Пароль

Повторите пароль

Имя

Фамилия

Электронная почта

ПРИНЯТЬ

ОТМЕНИТЬ

Рисунок 3.4 – Форма регистрации

После успешной авторизации интерфейс пользователя принимает вид в соответствии с ролью. Интерфейс системы представляет из себя набор корневых таблиц, отображающих списки основных сущностей – «Проекты», «Приложения», «Сценарии» и «Вычислители», доступ к которым осуществляется посредством панели навигации (см. рис. 3.5), а также специальные, соответствующие записи, страницы. Рассмотрим далее типовые сценарии использования системы для каждого из типов аккаунта. Отметим, что администратор системы имеет доступ к элементам управления, доступным для остальных ролей, а разработчику доступны возможности расчетчика.

Администратор системы отвечает за регистрацию новых вычислительных ресурсов, подготовку программного окружения и управление базой данных. При переходе на страницу «Вычислители», изображенную на рис. 3.5, администратор нажатием кнопки «Новая запись» получает доступ к форме регистрации вычислительного ресурса (см. рис. 3.6). К заполнению предлагаются следующие поля:

- «Имя» - название вычислительного ресурса, отображаемое в интерфейсе;
- «Тип доступа» - тип доступа к вычислительному ресурсу (в рамках прототипа – только SSH-соединение);
- «Адрес» - IP-адрес вычислительного ресурса или его доменное имя;
- «Порт» - порт для установления соединения;
- «Система управления очередью» - система управления заданиями пользователей, используемая на регистрируемом вычислительном ресурсе. Возможен выбор из двух вариантов – «native» (стандартный запуск приложений UNIX) и «SUPPZ» (СУППЗ);
- «Базовое окружение» - список переменных окружения.

После заполнения предложенной формы для регистрации

вычислительного ресурса в системе необходимо нажать кнопку «Принять».

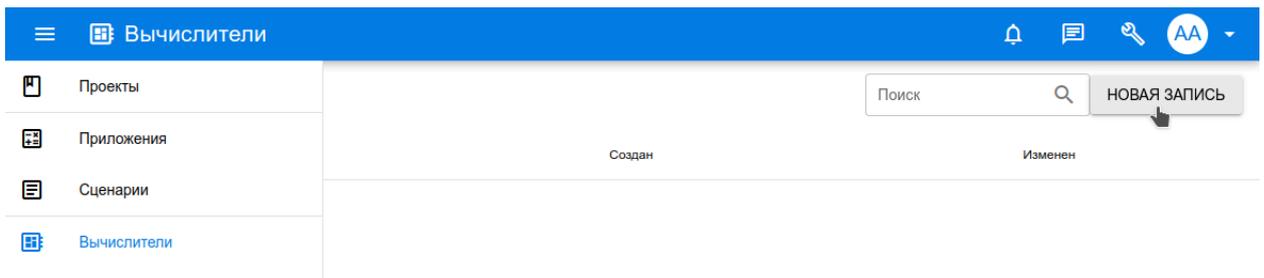


Рисунок 3.5 – Страница списка вычислительных ресурсов

Рисунок 3.6 – Форма регистрации нового вычислительного ресурса

При успешном создании записи пользователь будет перенаправлен на страницу вычислительного ресурса, изображенную на рис. 3.7. В рамках разработанного прототипа авторизация пользователя на удаленном вычислителе осуществляется с помощью RSA-ключа сервера, который необходимо добавить в соответствующий файл на удаленном вычислителе (в UNIX-системах обычно это файл `authorized_keys`, расположенный в `.ssh`, находящейся в домашней директории пользователя). Получить данный ключ возможно посредством нажатия кнопки «Показать SSH-RSA». После осуществления подготовки UNIX-пользователя на удаленном вычислителе

необходимо нажать кнопку «Создать доступ», в результате чего в интерфейсе отобразится форма создания доступа, изображенная на рис. 3.8. Поля, подлежащие заполнению, включают в себя «Имя пользователя» (имя пользователя на удаленном вычислителе) и «Рабочая директория» (директория, относительно которой будут создаваться директории проектов и расчетов, подготавливаемые системой), а также дополнительный список переменных окружения.

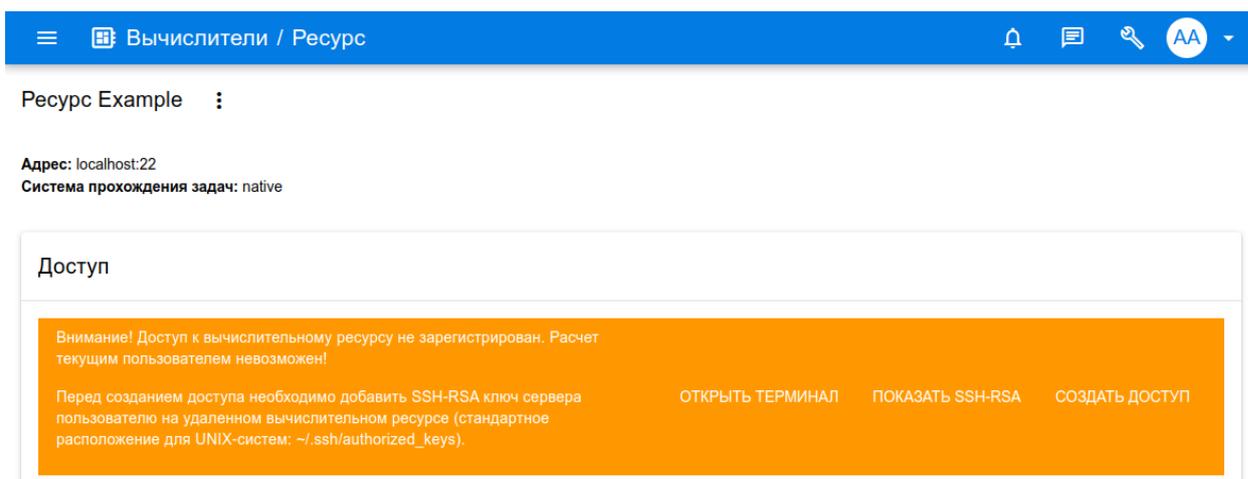


Рисунок 3.7 – Страница вычислительного ресурса

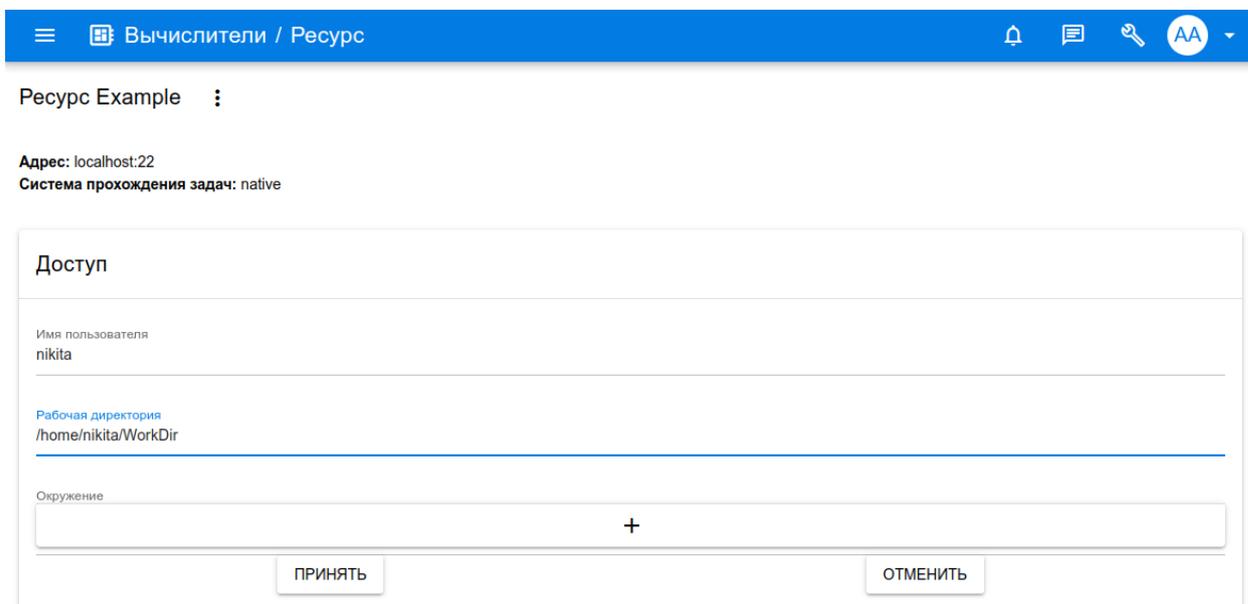


Рисунок 3.8 – Форма создания доступа

После завершения заполнения формы, нажатием кнопки «Принять», сервер проверит возможность соединения и корректность авторизации с

удаленным вычислительным ресурсом и, при успешном соединении, создаст записи типа «Access». Страница примет вид, подобный изображенному на рис. 3.9. В качестве дополнительного взаимодействия реализована возможность открытия SSH-терминала (см. рис. 3.10), посредством нажатия кнопки «Открыть терминал». Отметим, что перед использованием удаленного ресурса для проведения вычислительного эксперимента пользователь любой роли должен предварительно подготовить собственный доступ.

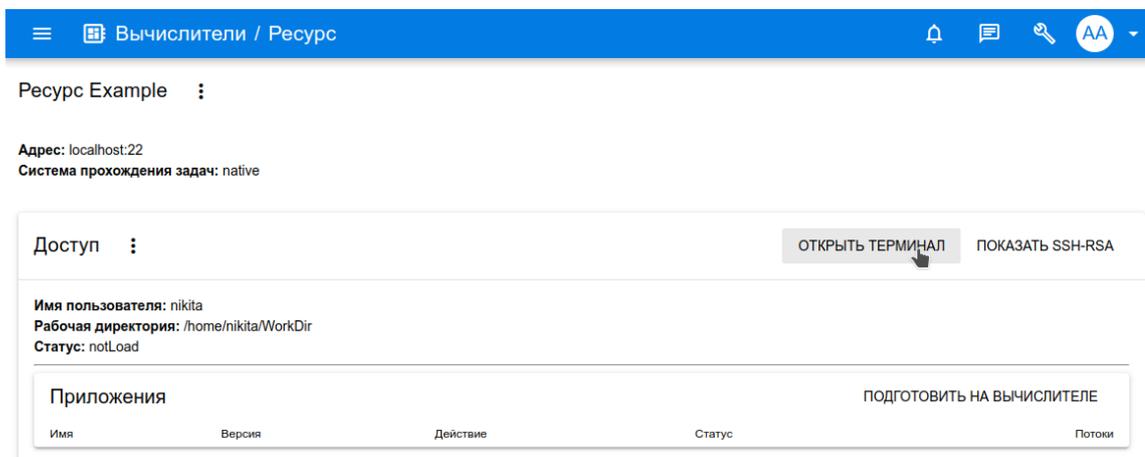


Рисунок 3.9 – Страница вычислительного ресурса с активным доступом

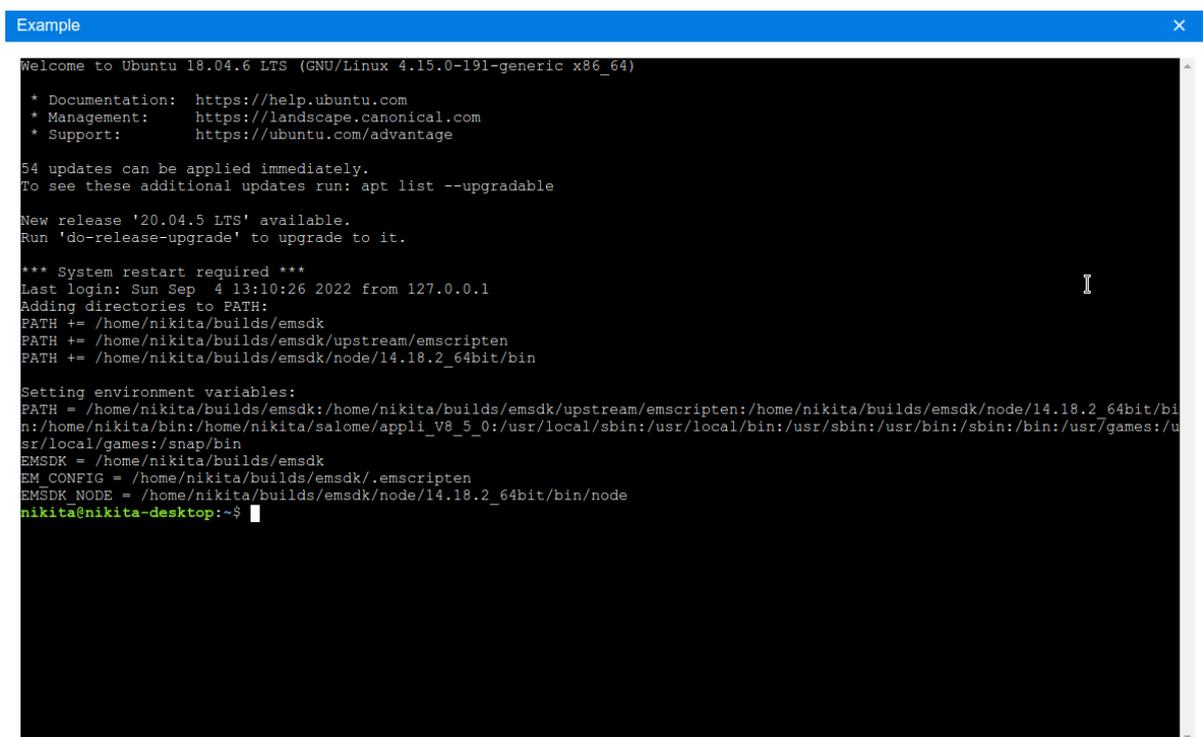


Рисунок 3.10 – Терминал вычислительного ресурса

Администратор системы, кроме возможности регистрации в системе новых вычислительных ресурсов, имеет возможность осуществить прямой доступ к таблицам базы данных посредством страницы, изображенной на рис. 3.11. Пособредством предлагаемого интерфейса пользователь может просмотреть, изменить или удалить записи из базы данных.

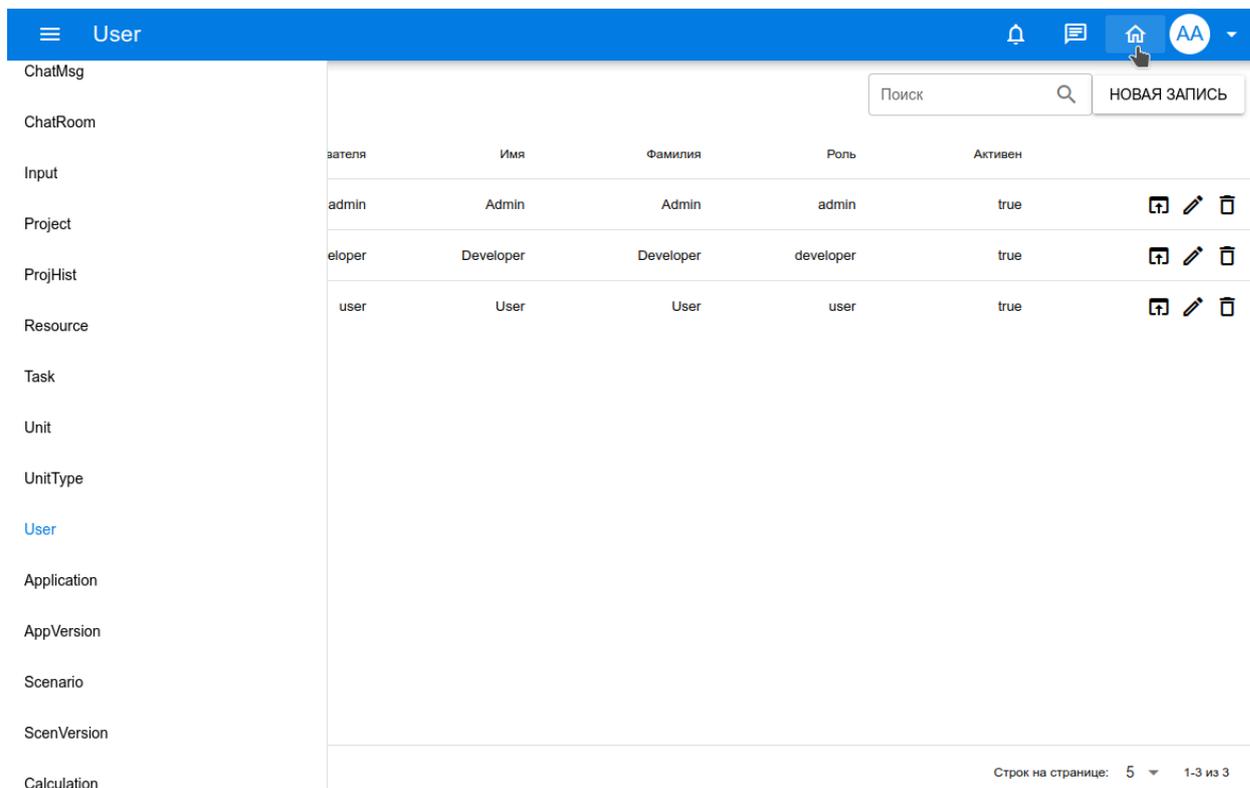


Рисунок 3.11 – Интерфейс доступа к базе данных

Основным сферой деятельности разработчика прикладного программного обеспечения являются формы регистрации приложений и сценариев на цифровой платформе. Обращаясь на страницу приложений, представленную на рис. 3.12, разработчик может инициировать процесс создания новой записи нажатием кнопки «Новая запись». После этого система отобразит форму создания нового приложения, изображенную на рис. 3.13. Полями, подлежащими заполнению являются:

- «Имя» - название приложения, отображаемое в интерфейсе пользователя;
- «Идентификатор» - уникальный идентификатор, посредством

которого производится обращение к приложению в сценарии запуска;

- «Публичный» - флаг, указывающий возможность доступа к приложению других пользователей системы;
- «Описание» - текстовое описание приложения, поддерживающее разметку в формате Markdown [128];
- «Шаблон паспорта» - шаблон функционального описания приложения, подставляется как значение по-умолчанию в форме создания новых версий приложения.

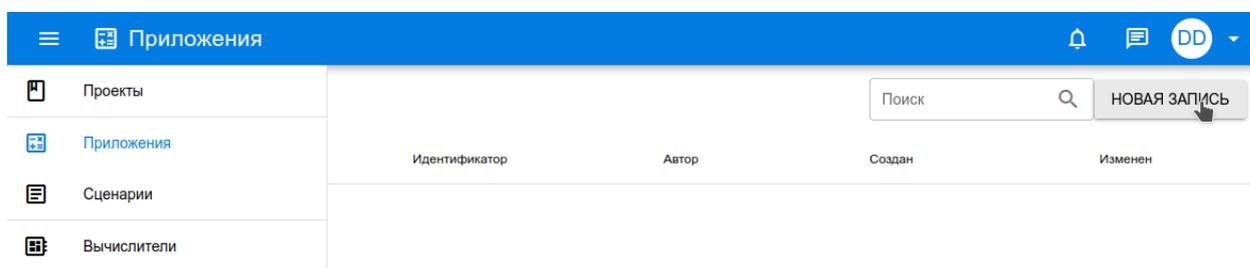


Рисунок 3.12 – Страница приложений

Рисунок 3.13 – Форма создания приложения

Нажатие кнопки «Принять» в форме отправит предоставленные пользователем данные на веб-сервер, который создаст новую запись в таблице базы данных, после чего пользователь будет перенаправлен на страницу приложения подобную, изображенной на рис. 3.14.

Нажатие кнопки «Новая запись» в таблице «Версии» перенаправит разработчика к форме создания новой версии приложения. Полями, подлежащими заполнению, являются:

- «Имя» - название версии приложения, используемое для отображения в интерфейсе;
- «Описание» - текстовое описание версии приложения;
- «Паспорт» - функциональное описание приложения, текстовый файл в формате YAML [129], содержащий такие элементы, как команды сборки и запуска, списки входных и выходных данных, их описание;
- «Исходный код» - архив, содержащий исходный код, описываемой версии приложения.

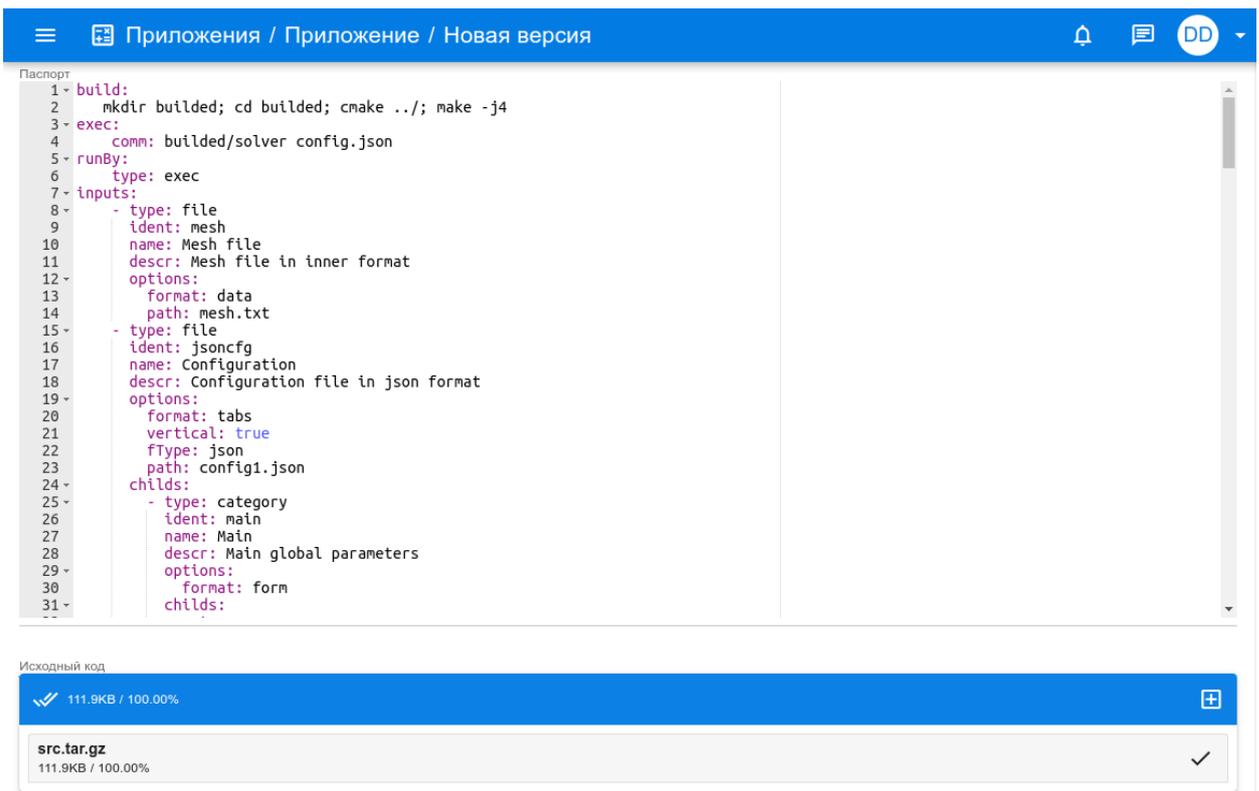


Рисунок 3.14 – Форма создания версии приложения

Для подготовки приложения на удаленном вычислительном ресурсе необходимо активировать вкладку «Подготовка», нажатие на которую отобразит таблицу «Ресурсы», изображенную на рис. 3.15 и содержащую

список вычислительных ресурсов, зарегистрированных в системе. В столбце версия пользователем определяется версия для подготовки. Последующее нажатие кнопки «Подготовить на вычислителях» запустит процесс отправки исходного кода на удаленный вычислительный ресурс с последующей сборкой исполняемого файла. Для доступа к стандартным потокам пользователю достаточно нажать кнопки «Stdout» или «Stderr». В процессе подготовки приложения столбцы «Действие» и «Статус» будут содержать описание текущего этапа и его состояние.

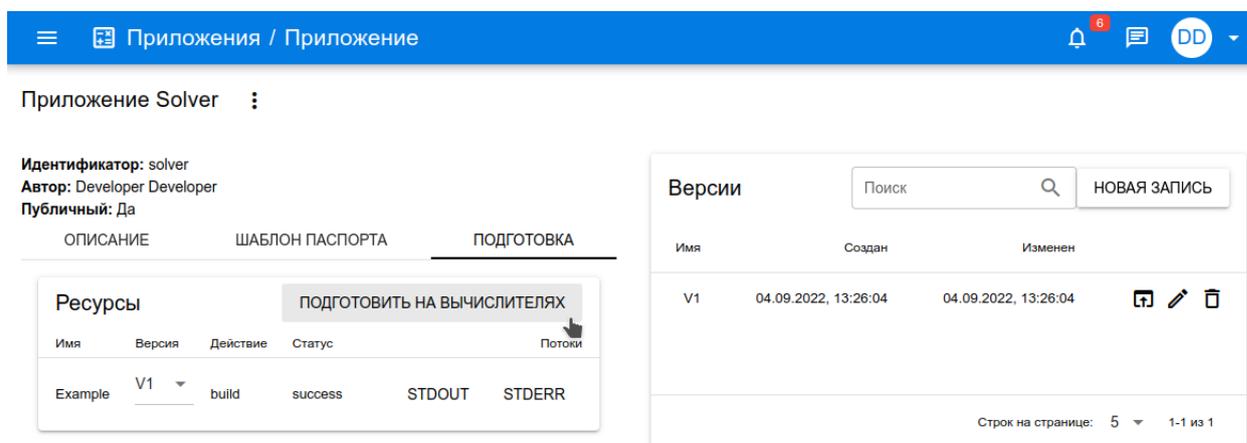


Рисунок 3.15 – Страница приложения. Подготовка на удаленном вычислительном ресурсе

Для создания сценариев на соответствующей странице, содержащей их список, необходимо нажать кнопку «Новая запись». Система предложит разработчику заполнить форму, изображенную на рис. 3.16 и содержащую поля:

- «Имя» - название сценария, отображаемое в системе;
- «Публичный» - флаг, отмечающий доступность сценария для других пользователей;
- «Описание» - текстовое описание сценария, поддерживающее разметку Markdown;
- «Шаблон скрипта» - используется для подстановки в поле «Скрипт» версии сценария как значение по умолчанию.

Сценарии / Новый сценарий

Новый сценарий

Имя  
Сценарий решателя

Публичный

Описание

1 - ## Сценарий решателя
--------------------------

Сценарий решателя

Рисунок 3.16 – Форма создания сценария

После окончания работы с формой необходимо нажать кнопку «Принять» для завершения ввода. При успешной отправке данных на сервер и создания записи, пользователь будет перенаправлен на страницу сценария. Нажатие кнопки «Новая запись» в таблице «Версии» перенаправит разработчика к форме создания новой версии сценария, изображенной на рис. 3.17. Данная форма содержит поля:

- «Имя» - название версии сценария;
- «Описание» - текстовое описание версии сценария в формате Markdown;
- «Скрипт» - функциональное описание сценария в формате YAML, содержащее ключи: «applications» - словарь приложений, где в качестве ключа используется строка, обозначающая приложение в рамках сценария, а в качестве значения – идентификатор приложения; «connections» - связи входных и выходных данных приложений; «stages» - этапы запуска приложений.

Сценарии / Сценарий / Новая версия

DD

ПРОВЕРИТЬ СКРИПТ

Скрипт

```
1- applications:
2-   hgSolver: solver
3- connections: []
4- stages:
5-   - run(hgSolver)
6-
7-
```

ПРИНЯТЬ

ОТМЕНИТЬ

Рисунок 3.17 – Форма создания версии сценария

Нажатие кнопки «Проверить скрипт» отобразит шаблон расчета, изображенный на рис. 3.18, позволяющий проверить корректность YAML-файла, существование приложений с указанными идентификаторами в базе данных, корректность форм входных и выходных данных. После окончания работы с формой для подтверждения ввода и отправки данных на сервер достаточно нажать кнопку «Принять».

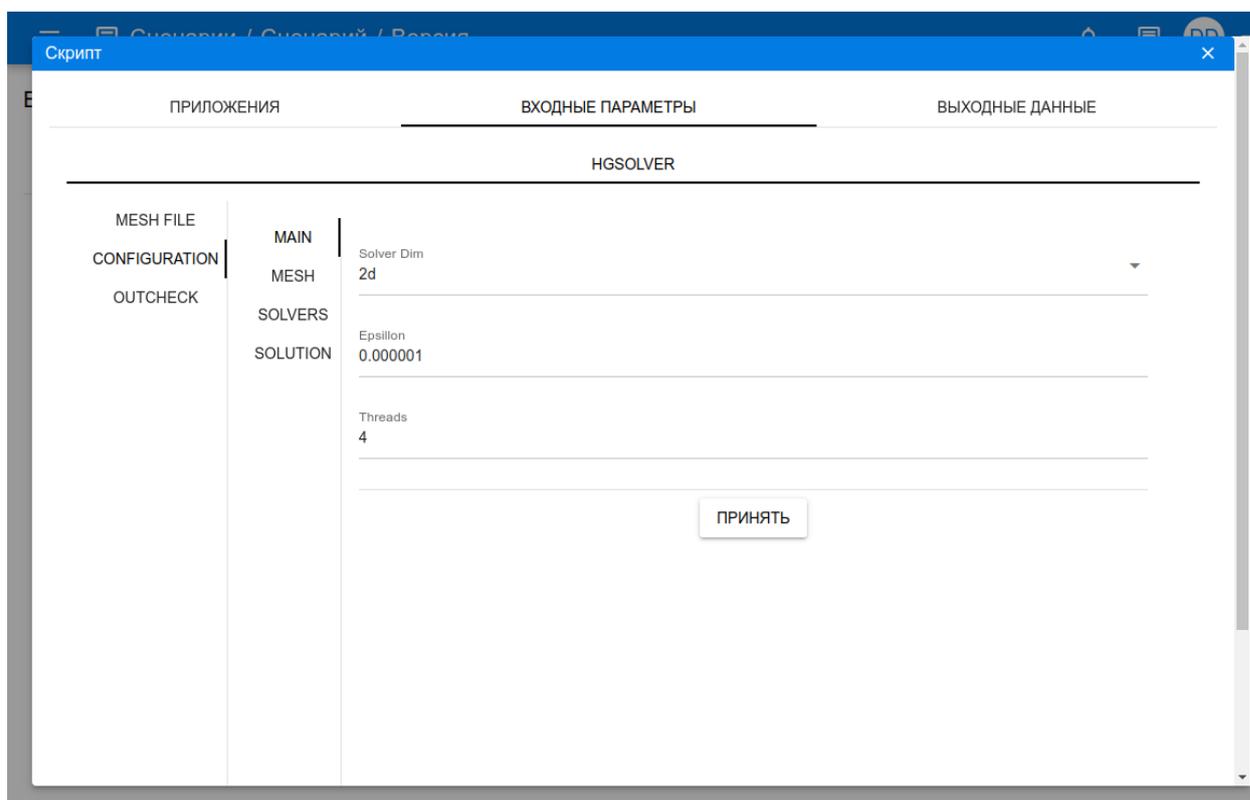


Рисунок 3.18 – Проверка корректности скрипта

Основной задачей пользователя с ролью расчетчик является проведение вычислительных экспериментов. В рамках разработанного прототипа данный процесс реализуется следующим образом. Нажатие на кнопку «Новая запись» на странице, содержащей список проектов, направит пользователя на страницу создания проекта, имеющую форму с полями (см. рис. 3.19):

- «Имя» - название проекта;
- «Описание» - текстовое описание проекта в формате Markdown;
- «Участники» - список пользователей – участников проекта.

После заполнения предложенных полей для передачи информации на сервер и создания записи в базе данных необходимо нажать кнопку «Принять». В результате система предложит страницу проекта, подобную изображенной на рис. 3.20.

Проекты / Новый проект

Новый проект

Имя  
Проект

Описание

1 - ### Цели	Цели
2	
3	
4 - ### Задачи	Задачи
5	
6	

Рисунок 3.19 – Форма создания проекта

Проекты / Проект

Проект Проект :

Участники: User User

ОПИСАНИЕ      ИСТОРИЯ

Цели

Задачи

Расчеты		
Поиск		НОВАЯ ЗАПИСЬ
Имя	Создан	Изменен
Нет данных		

Рисунок 3.20 – Страница проекта

Для создания нового расчета в таблице «Расчеты» необходимо нажать кнопку «Новая запись». Система перенаправит пользователя на страницу, содержащую форму создания нового расчета, изображенную на рис. 3.21. Данная форма содержит поля:

- «Имя» - название расчета;

- «Описание» - текстовое описание расчета в формате Markdown;
- «Сценарий» и «Версия» - выпадающие списки для выбора сценария и его версии.

После завершения работы с формой для регистрации в системе нового расчета достаточно нажать кнопку «Принять». После успешного создания записи пользователь будет направлен на страницу, подобную изображенной на рис. 3.22.

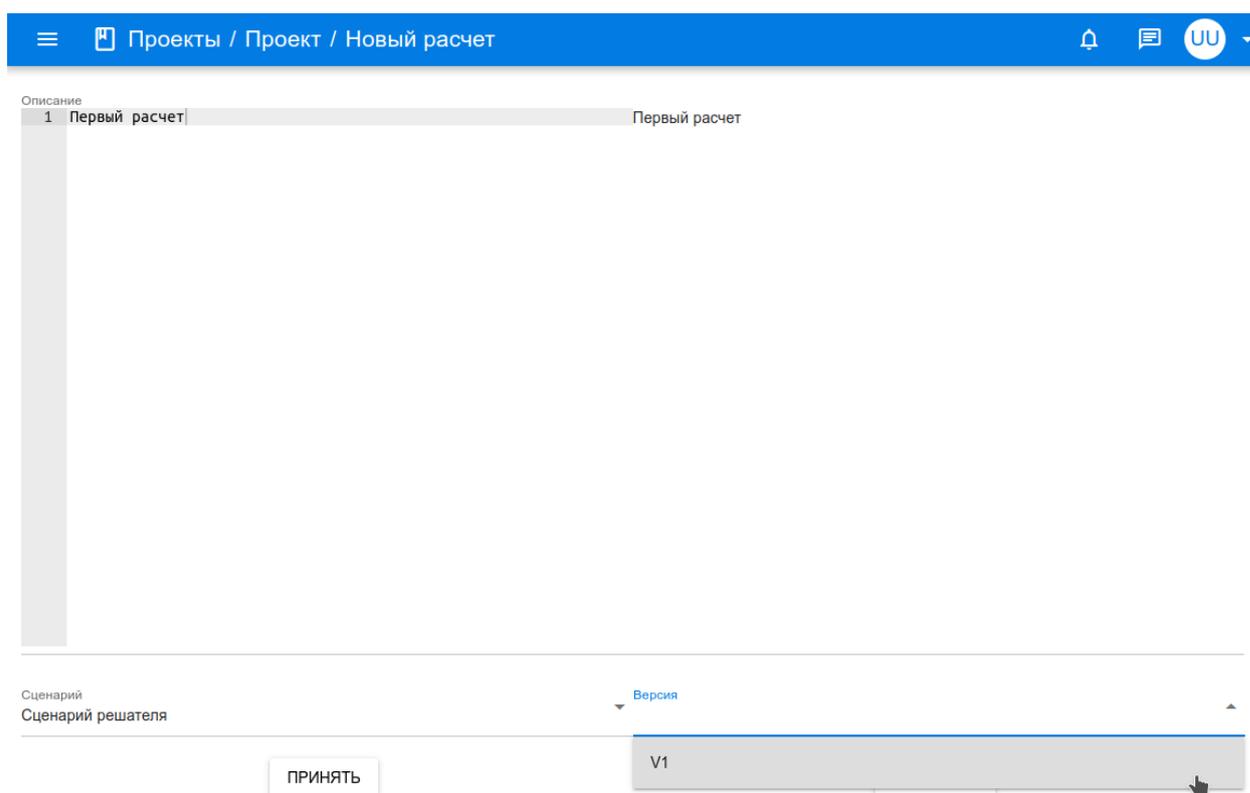


Рисунок 3.21 – Форма создания расчета

В рамках страницы расчета происходит основное взаимодействие с удаленным вычислительным ресурсом, включающее управление и слежение за пользовательскими процессами, подготовку исходных данных и анализ выходных. На вкладке «Вычислители» пользователь может определить, на каком из вычислительных ресурсов будет происходить запуск приложения, из списка указанных в сценарии. Активация «Входные параметры» отобразит формы в соответствии с описанием, указанным в паспорте приложения. При условии, что входным параметром являются файл данных или директория,

системой будет отрисована форма, изображенная на рис. 3.22. посредством данной формы возможно либо отправить файл, либо связать с результатами расчетов, полученными в системе ранее. Если параметры являются числовыми, строковыми, списками или иными поддерживаемыми типами, оснащенными компонентами система предложит комплексную веб-форму, например, изображенную на рис. 3.23. Подтверждение ввода производится нажатием кнопки «Принять».

Рисунок 3.22 – Задание входных данных

Рисунок 3.23 – Задание параметров расчета

После завершения подготовки входных параметров возможно произвести запуск на расчет. Для этого необходимо активировать вкладку «Запуск», содержащую компонент управления расчетным процессом (см. рис. 3.24). В таблице «Приложения» в зависимости от типа запуска и системы прохождения пользовательских заданий, столбец «Параметры» будет содержать необходимые для запуска поля (например –pr для процессов, запускаемых с помощью `mpirun`). Управление (запуск, останов) приложениями осуществляется посредством таблицы «Этапы» с помощью нажатия соответствующих кнопок. При этом актуальный статус задачи будет указан в столбце «Статусы». Для просмотра стандартных потоков вывода и ошибки достаточно нажать кнопки «Stdout» и «Stderr» соответственно (см., например, рис. 3.25).

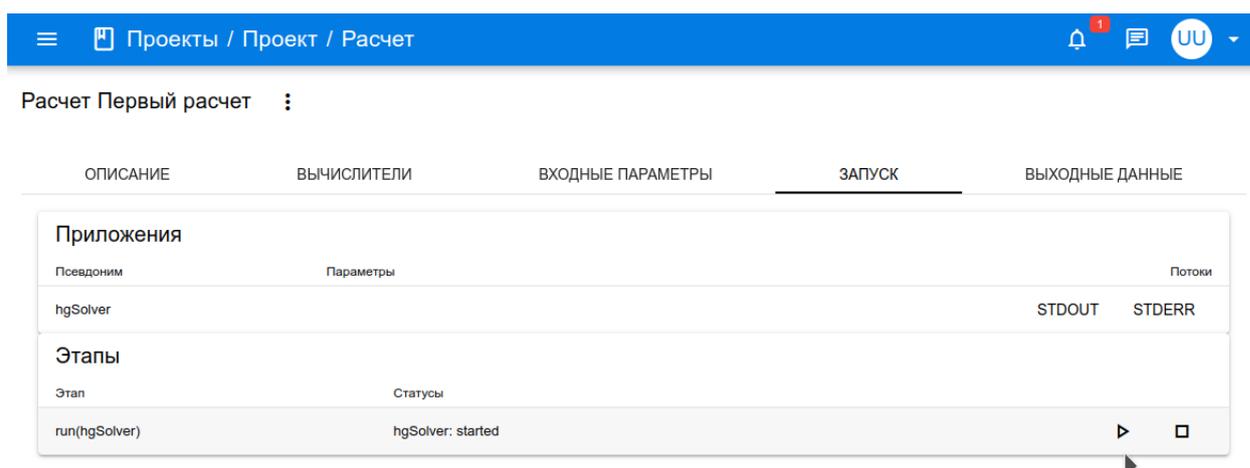


Рисунок 3.24 – Компонент управления расчетным заданием

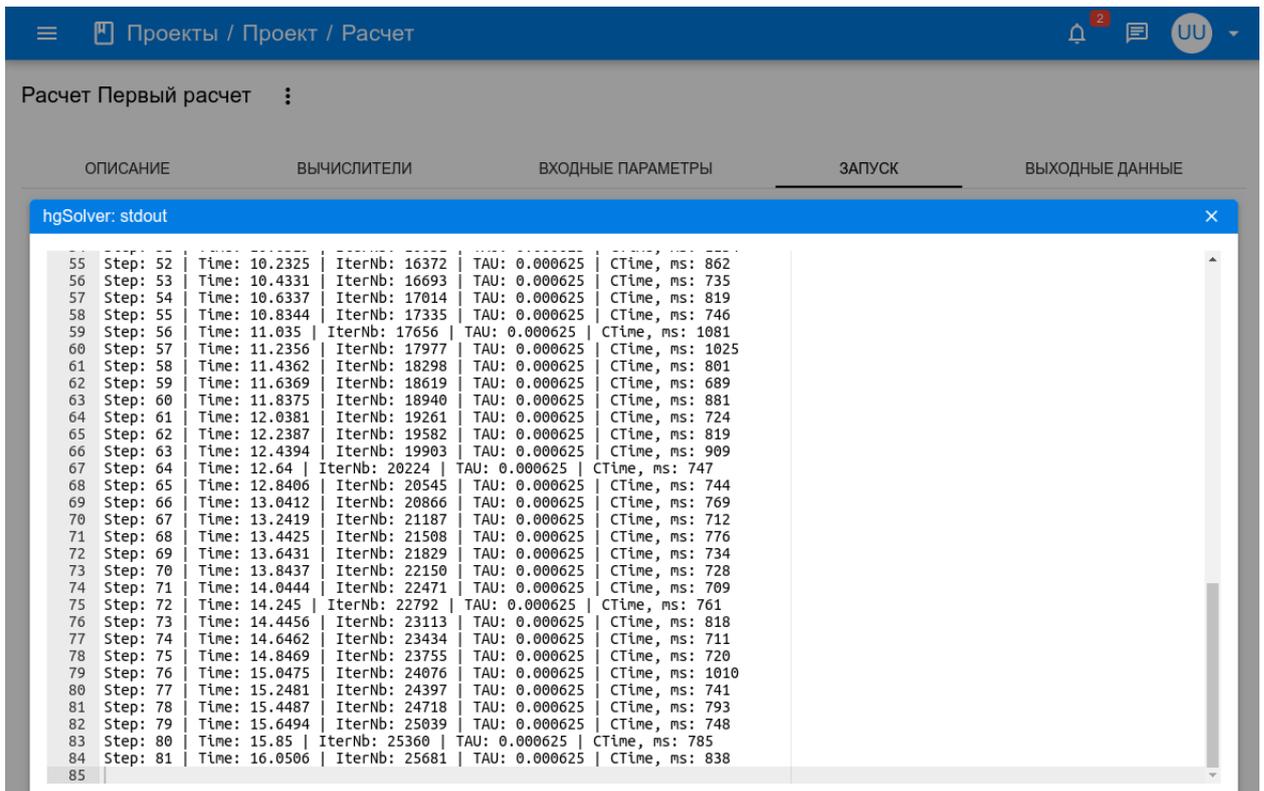


Рисунок 3.25 – Просмотр стандартного потока вывода

Для анализа выходных данных пользователю необходимо активировать вкладку «Выходные данные», в результате чего система представит компонент пользовательского интерфейса, подобный изображенному на рис. 3.26. В левой части компонента будет сформирована древовидная структура, содержащая списки выходных данных. Для директорий предусмотрена возможность отображения списка содержащихся в них файлов. При активации элемента списка справа будет отображаться информация о выходных данных, включая тип, описание, время изменения, размер и тип. Также пользователь может воспользоваться такими функциями, как «Открыть», «Скачать» или «Очистить».

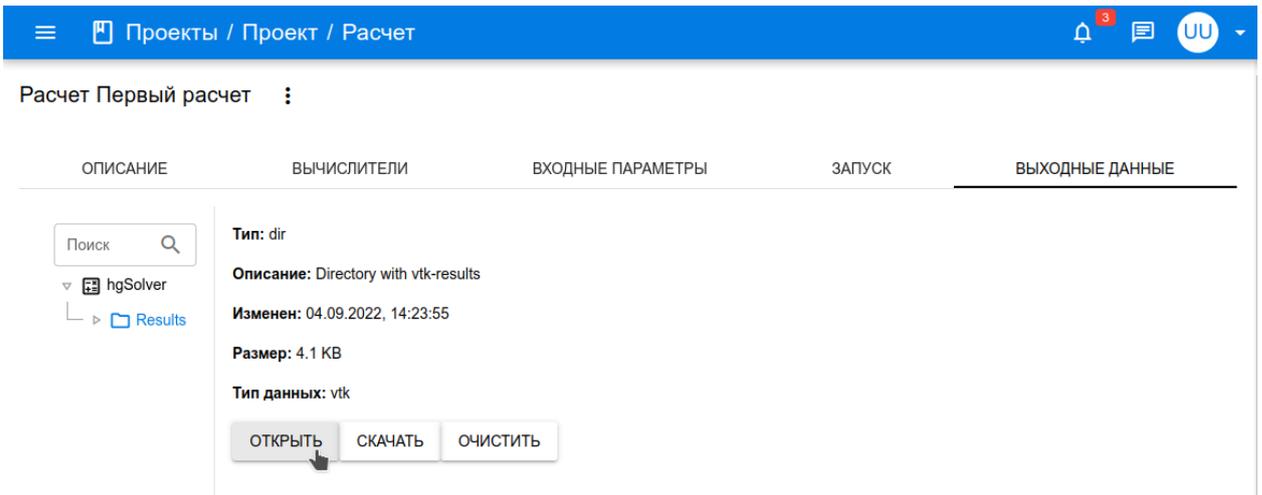


Рисунок 3.26 – Список выходных данных

Поведение системы при нажатии кнопки «Открыть» зависит от типа активных выходных данных. Например, при открытии типа «vtk» система отобразит компонент, содержащий Visualizer.js, пример которого изображен на рис. 3.27. В рамках прототипа реализованы возможности непосредственного просмотра растровых изображений и видеозаписей, текстовых файлов и данных библиотеки VTK.

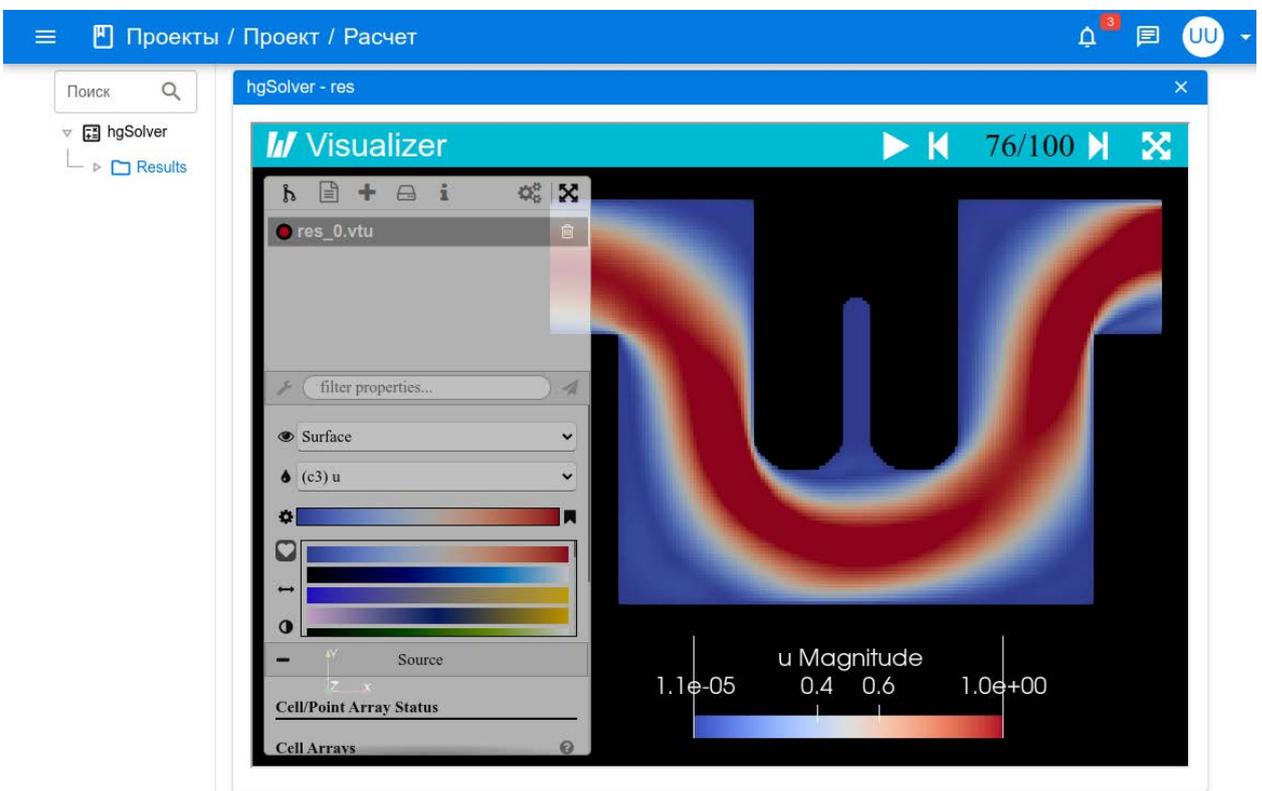


Рисунок 3.27 – Просмотр файлов визуализации

### **3.5 Выводы по Главе 3**

В третьей главе рассмотрены актуальные направления развития суперкомпьютерных и кластерных вычислителей и обсуждены существующие подходы организации разделения вычислительных ресурсов.

Предложена клиент-серверная архитектура цифровой платформы для проведения полного цикла вычислительного эксперимента, предназначенная в том числе для обеспечения графического пользовательского интерфейса расчетным кодам, представленным в главе 2.

Установлен стек технологий, на базе которых реализован прототип веб-системы. Представлен типичный сценарий использования цифровой платформы для каждого из типов пользовательского аккаунта.

Результаты данной главы опубликованы в работе [35].

## Глава 4. Вычислительные эксперименты

### 4.1 Задачи верификации

#### 4.1.1 Установление течения Пуазейля

Численный расчет установления течения Пуазейля проводился для числа Рейнольдса равного 100 от начальных условий (1.31). Использовались три различных декартовых сетки:

- 1) линейный размер элемента – 0.05 (количество элементов – 1 600);
- 2) линейный размер элемента – 0.025 (количество элементов – 6 400);
- 3) линейный размер элемента – 0.01 (количество элементов – 40 000).

Результаты численного расчета для третьей сетки приведены на рис. 4.1 и 4.2.

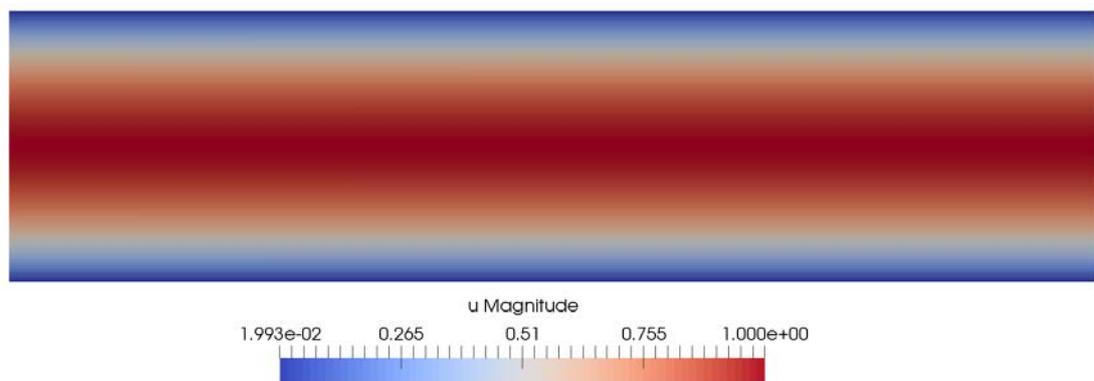


Рисунок 4.1 – Распределение модуля скорости

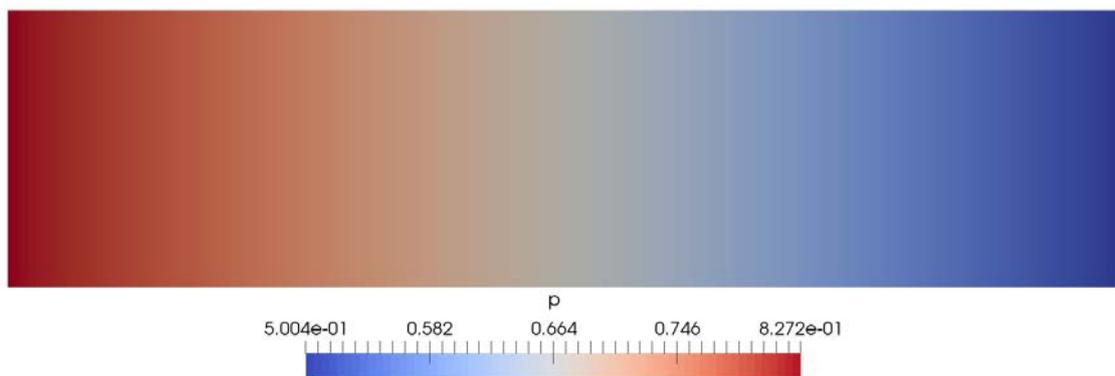


Рисунок 4.2 – Распределение давления

Распределение продольной скорости на поперечном сечении в центре области ( $X=2$ ), представлено на рис. 4.3. Пунктирной линией изображена аналитическая зависимость:

$$u_x(y) = 1 - \frac{(y - 0.5)^2}{0.25}.$$

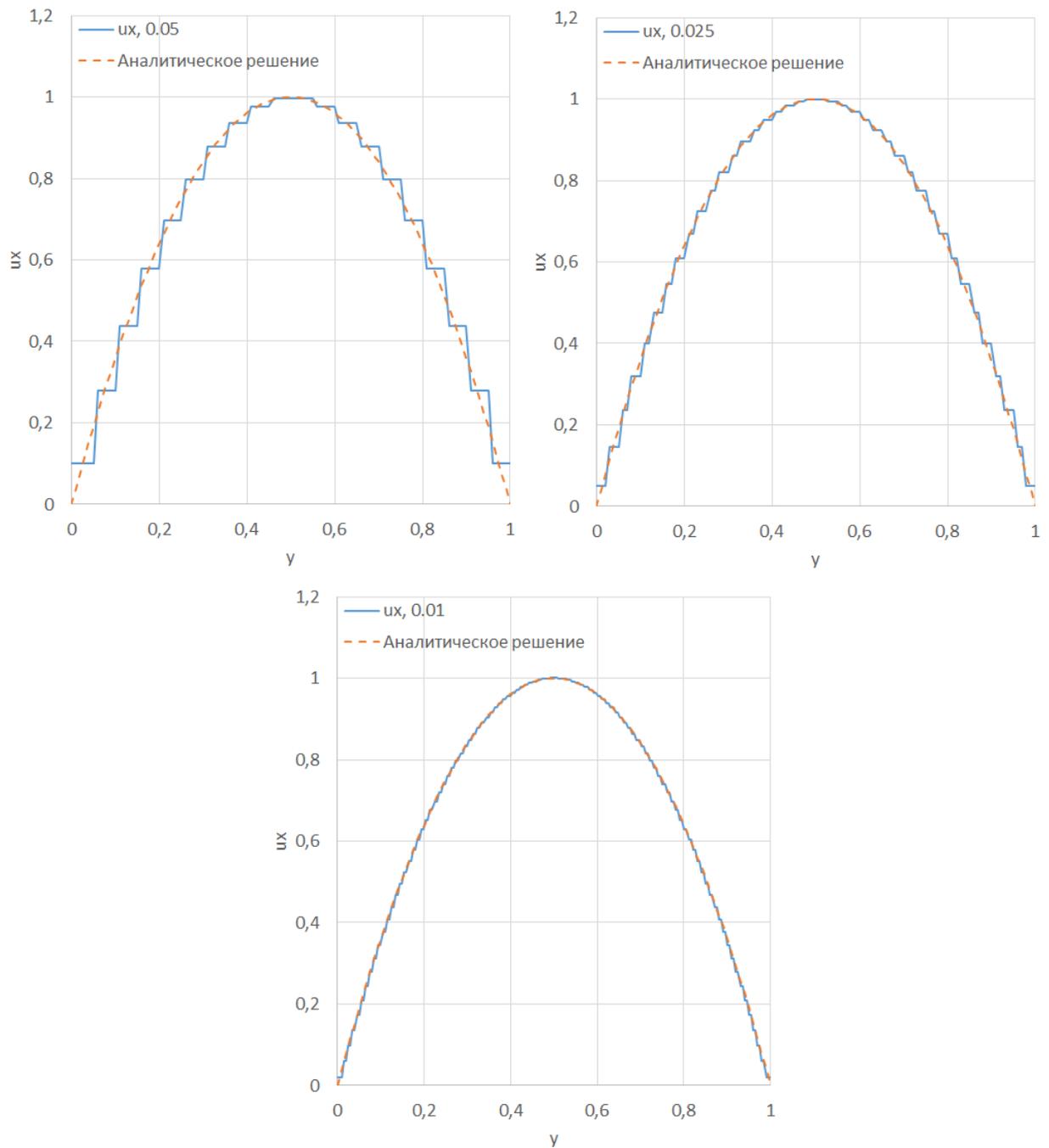


Рисунок 4.3 – Распределения продольной скорости при  $X=2$  для сеток: 0.05 – слева сверху; 0.025 – справа сверху, 0.01 – снизу

Как видно из представленных результатов течение Пуазейля установилось в исследуемой области. Кроме того, с уменьшением шага сеточного элемента наблюдается сходимость к точному решению.

#### 4.1.2 Перенос концентрации

Для проверки реализации программного модуля, отвечающего за численный расчет уравнений конвекции-диффузии, рассмотрим квадратную область  $2 \times 2$ . В качестве начальных данных будем использовать следующие выражения:

$$u_x = 0.5, \quad u_y = 0.5, \quad p = 0.5, \quad C(x, y) = \exp\left[-\frac{(x - 0.5)^2 + (y - 0.5)^2}{D_0}\right].$$

Коэффициент диффузии будем считать равным  $D_0 = 0.005$ . Отметим, что пересчет скоростей и давления от начального состояния не производился. Численный эксперимент проводился на декартовой сетке с линейным размером элемента равным 0.01, состоящей из 40 000 квадратов. Эволюция распределения концентрации представлена на рис. 4.4 и рис. 4.5.

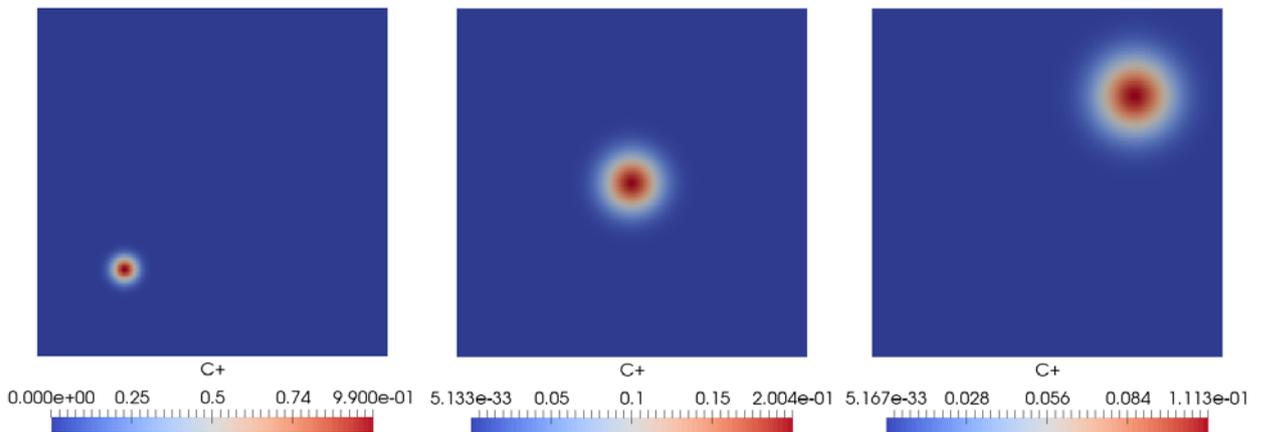


Рисунок 4.4 – Распределение концентрации при  $t = 0.0$  (слева),  $t = 1.0$  (по центру) и  $t = 2.0$  (справа)

Аналитическое решение данной задачи имеет вид [73]:

$$C(x, y, t) = \frac{1}{4t+1} \exp \left[ -\frac{(x-0.5-u_x t)^2 + (y-0.5-u_y t)^2}{D_0(4t+1)} \right].$$

Графики, изображенные на рис. 4.6, показывают хорошее соответствие результатов численного расчета аналитическому решению как по амплитуде, так и по скорости перемещения пика.

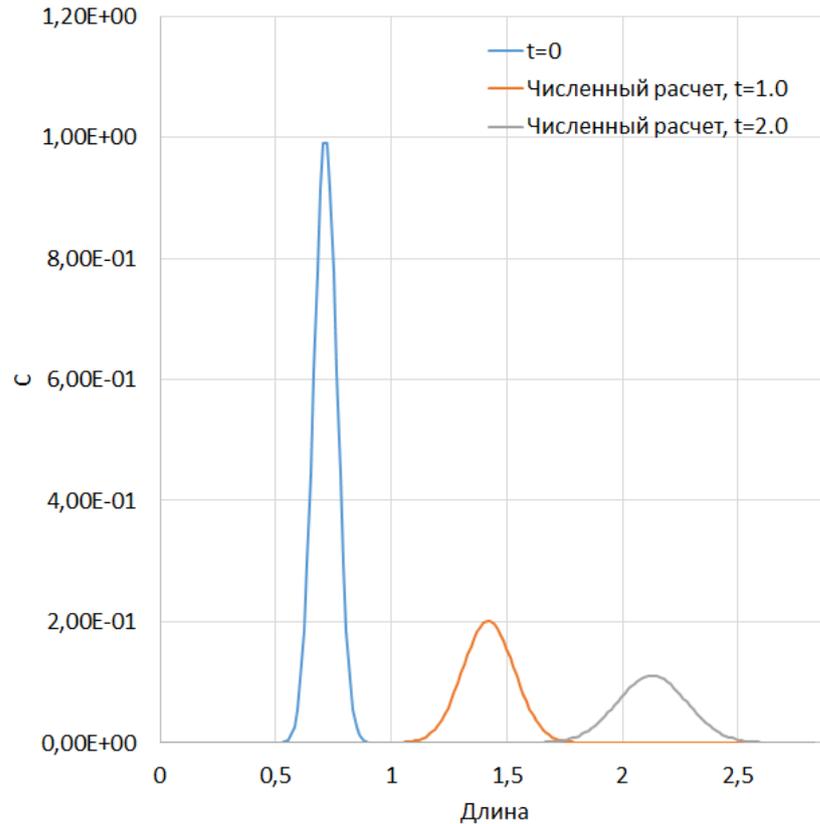


Рисунок 4.5 – Эволюция распределения концентрации вдоль линии от {0,0} до {2,2}

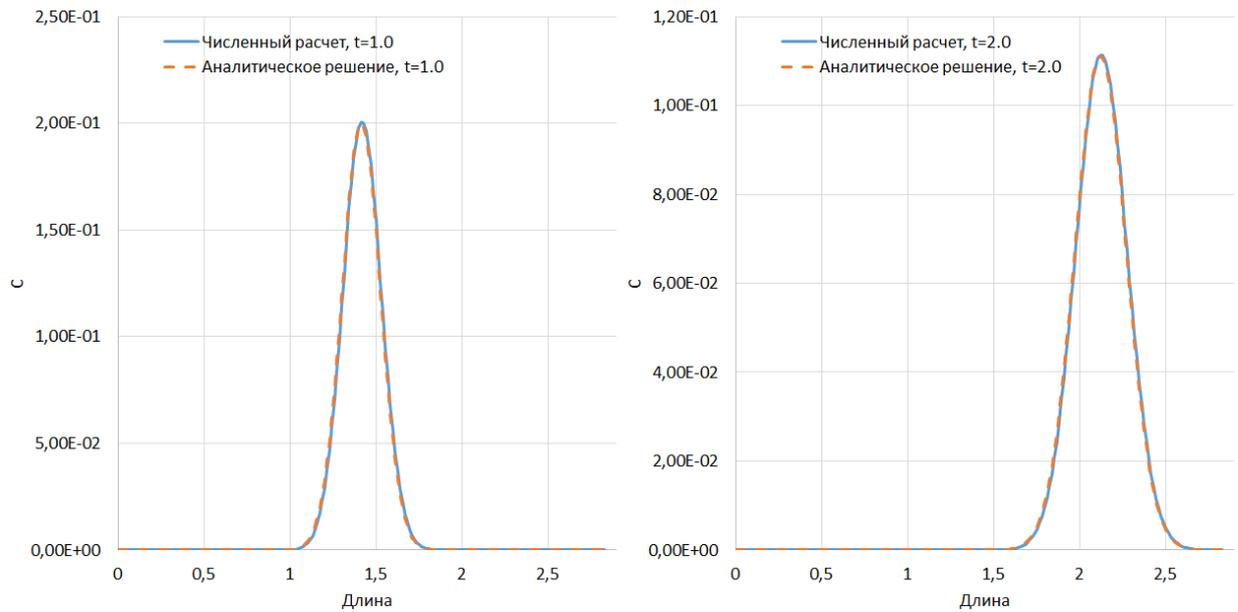


Рисунок 4.6 – Численный расчет и аналитическое решение при  $t = 1$  (слева) и  $t = 2$  (справа)

### 4.1.3 Тепловое конвективное движение

Рассмотрим движение, порождаемое разностью температур в замкнутом резервуаре с размерами  $1 \times 1$ . Граничные условия КГиД-блока соответствуют (1.30), начальные – (1.31). По температуре воспользуемся условиями (1.30) на верхней и нижней границах, имея в виду отсутствие теплообмена, на левой границе примем  $T = 1$ , на правой –  $T = 0$ , в качестве начальных данных возьмем нулевое распределение (1.35).

Параметры расчета:  $Re = 100$ ,  $Pe = 10$ . Вычислительные эксперименты будем проводить для чисел Грасгофа:  $Gr = 1, 10, 100$  на декартовой сетке с характерным линейным размером элемента 0.01 (количество сеточных элементов – 10 000). Результаты модельных расчетов приведены на рис. 4.7.

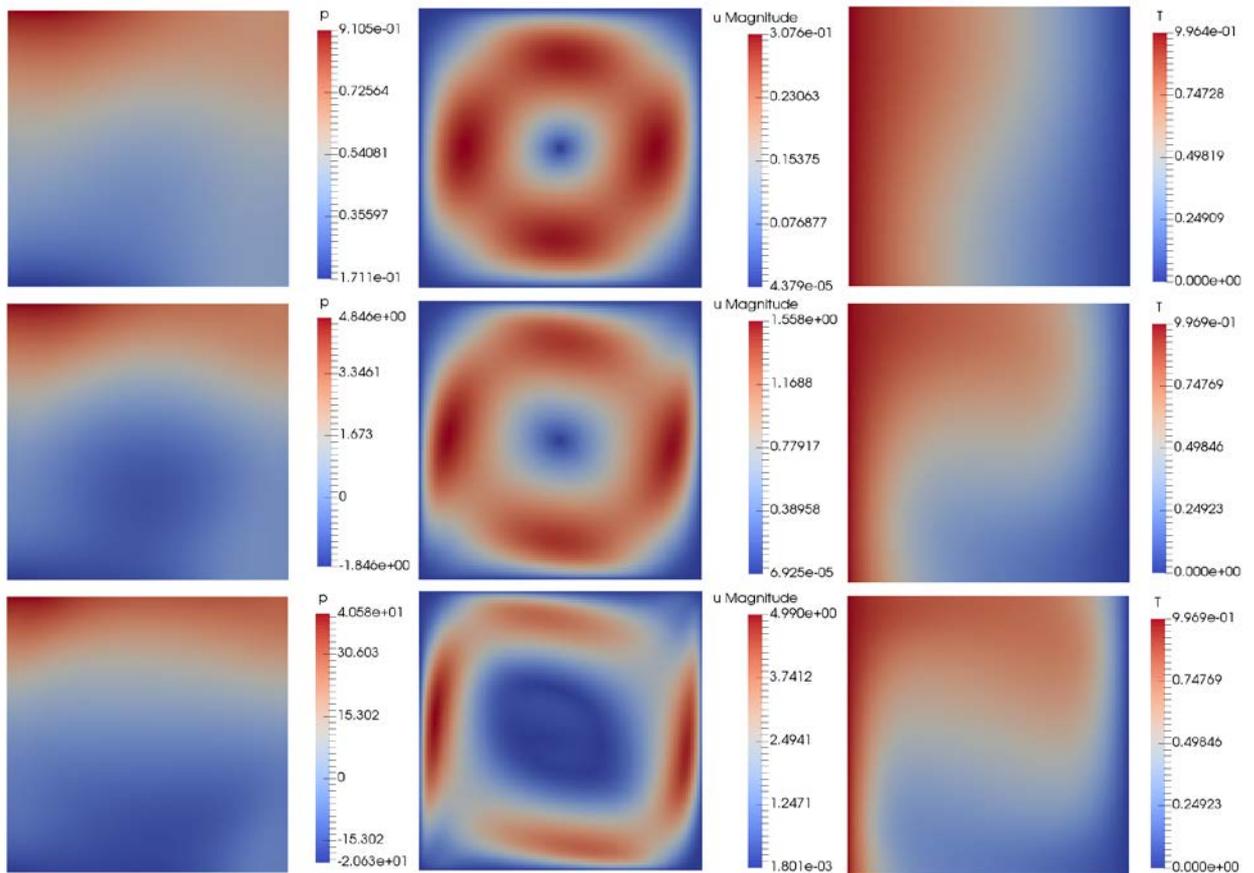


Рисунок 4.7 – Результаты модельного расчета. Слева – давление, в центре – модуль скорости, справа – температура. Число Грасгофа сверху вниз соответственно 1, 10, 100

## 4.2 Задача валидации

В качестве задачи валидации рассматривался безмембранный метод обессоливания морской воды, описанный в работе [74], заключающийся в геометрическом расщеплении потока и локализацией растворенных ионов посредством анода. Моделирование проводилось для вырезки центра размером 0.1 см. При проведении численного эксперимента на входном отверстии задавалось течение Пуазейля (в качестве радиуса использовался радиус описанной окружности). На выходе задавались мягкие граничные условия, на стенках – прилипание. Число Рейнольдса принималось равным 100. Расчет проводился на гексаэдральной сетке, содержащей 78 200 элемента (характерный линейный размер – элемента 0.05). Стационарное течение, полученное в результате моделирования, изображено на рис. 4.8.

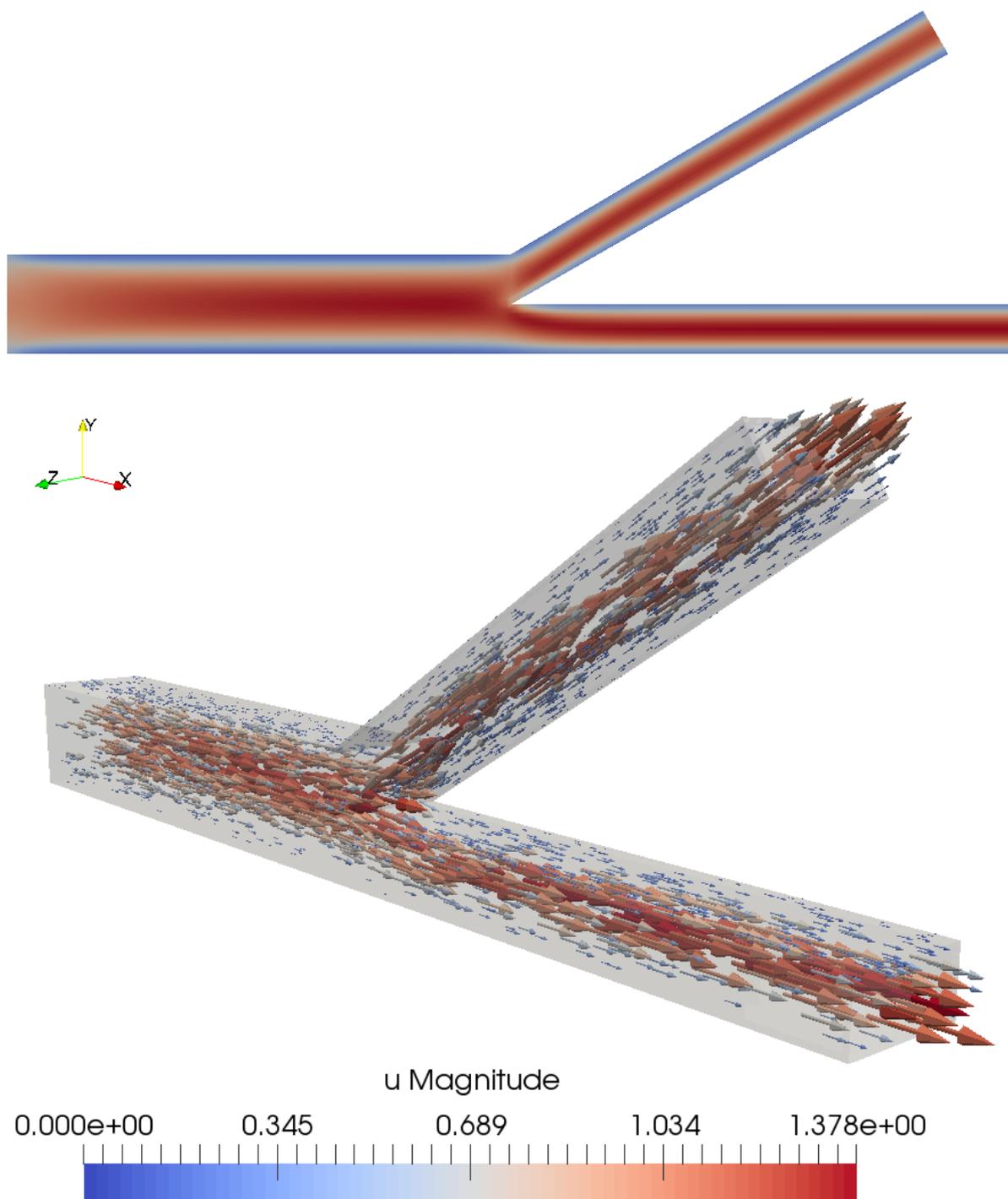


Рисунок 4.8 – Распределение модуля скорости в сечении  $Z=0$  (сверху) и в объеме (снизу)

Моделирование эволюции концентрации проводилось при коэффициентах диффузии для обоих компонентов смеси  $D_+ = D_- = 0.005$ , при одинаковом воздействии силы Лоренца  $F_+ = -F_- = 1.0$ . Начальные значения положительно и отрицательно заряженных ионов принимались равными

нулю, на входе задавалось значение  $C_+ = C_- = 0.5$ . Потенциал электрического поля на электроде принимался  $\varphi = 1.0$ , на входе равным нулю. В результате численного расчета концентрации вышли на стационарный режим, полученные результаты для суммы концентраций представлены на рис. 4.9.

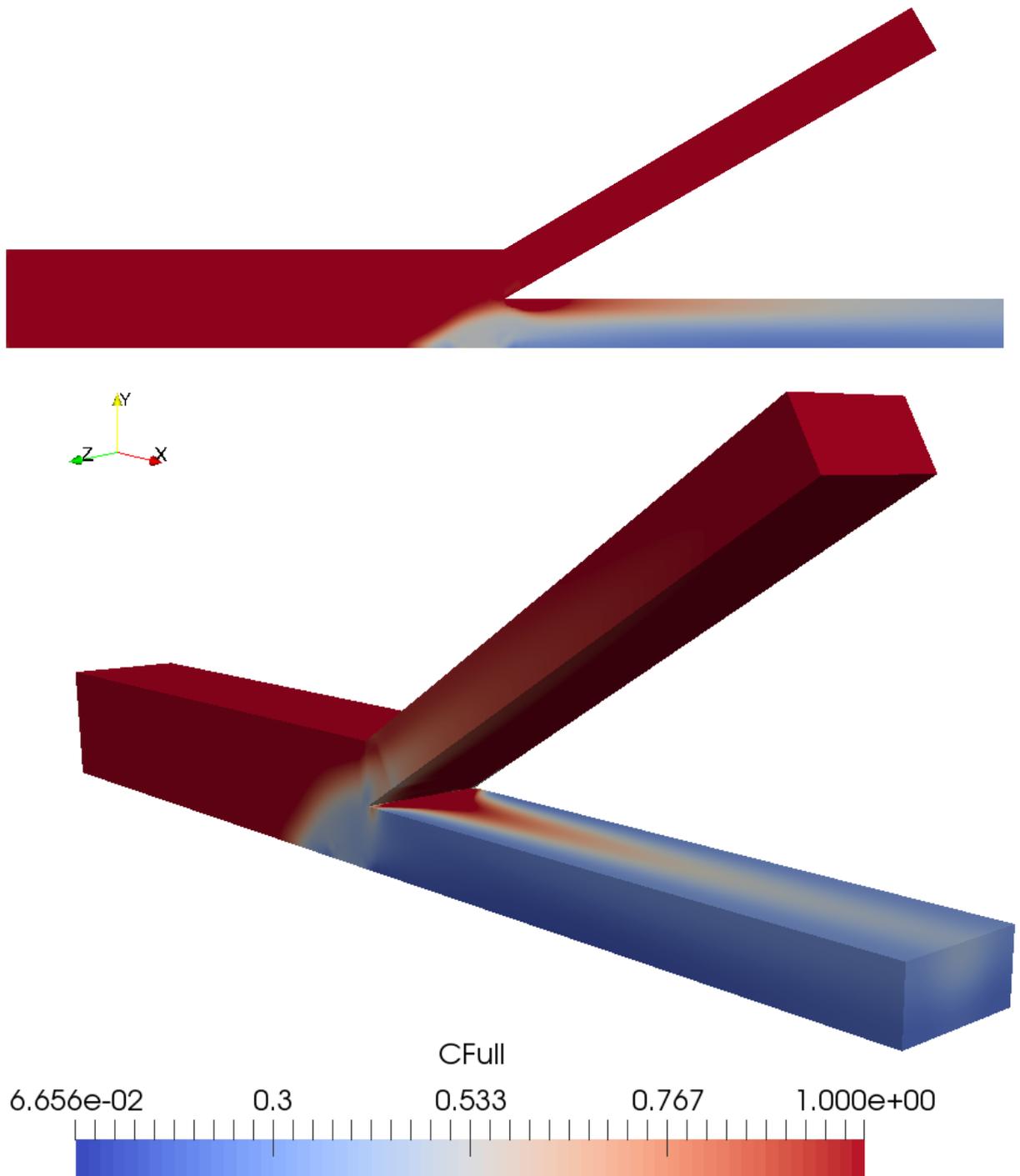


Рисунок 4.9 – Распределение концентрации в сечении  $Z=0$  (сверху) и на поверхности (снизу)

Общий характер стационарного течения и распределения загрязнителя согласуется с приведенным в работе [74], насыщенная солью жидкость протекает через верхний патрубок. Обессоленная вода протекает по нижнему.

### 4.3 Модельные задачи

#### 4.3.1 Задача об электромагнитной очистке водной среды

Расчеты проводились на сетке, состоящей из треугольных призм, содержащей 131 296 элементов. При решении задачи использованы стандартные граничные условия, описанные в главе 1. Полученные для  $Re = 100$  распределения параметров потока представлены на рисунках 4.10 – 4.13.

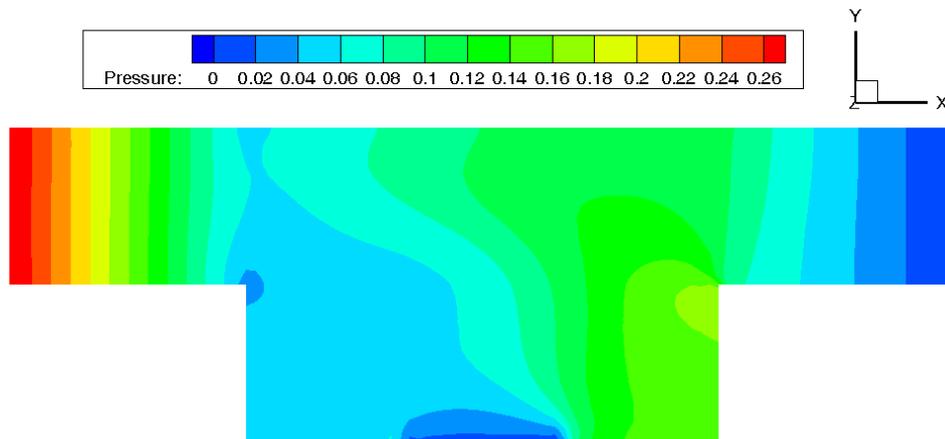


Рисунок 4.10 – Распределение давления в центральном сечении

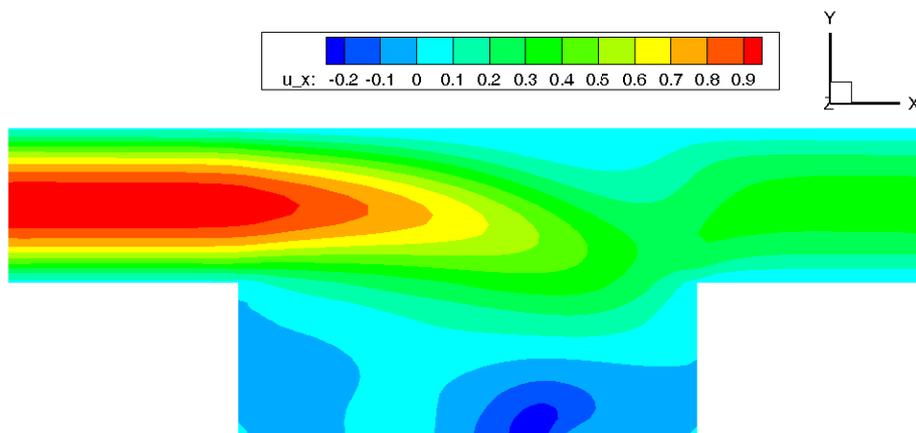


Рисунок 4.11 – Распределение продольной скорости в центральном сечении

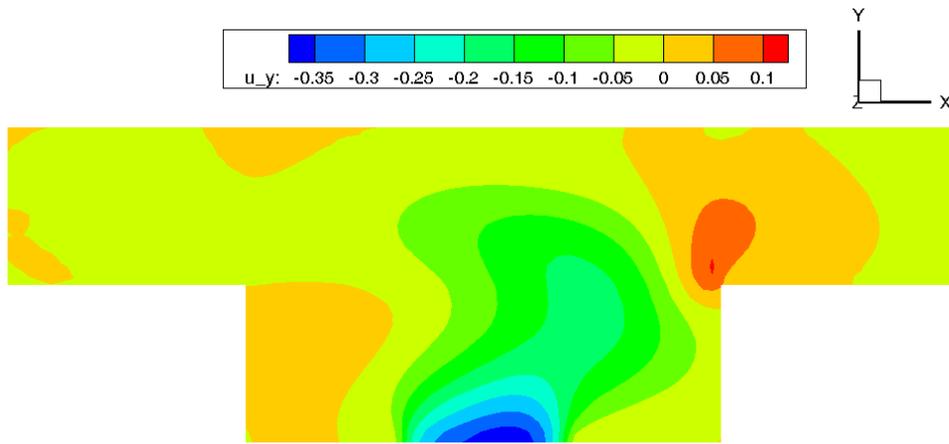


Рисунок 4.12 – Распределение поперечной скорости в центральном сечении

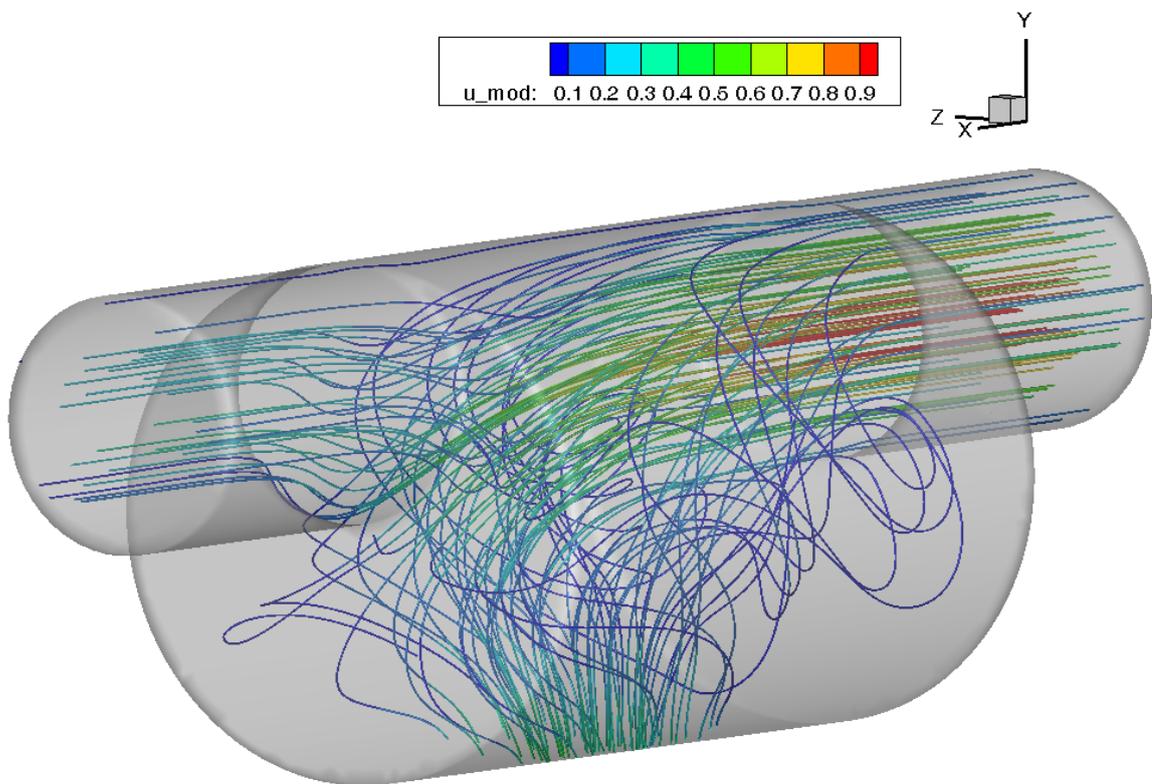


Рисунок 4.13 – Линии тока, общий вид

В качестве примеси рассматривались положительно заряженные ионы, растворенные в жидкости. Расчет распределения концентрации примеси проводился для нормировочных коэффициентов  $D_0 = 1.0$ ,  $B_0 = 3.0$ ,  $\varphi_0 = 1.0$ . В результате удалось добиться стационарного состояния, представленного на рисунках 4.14, 4.15. На рисунках также виден эффект очистки верхней части потока от положительно заряженной примеси.

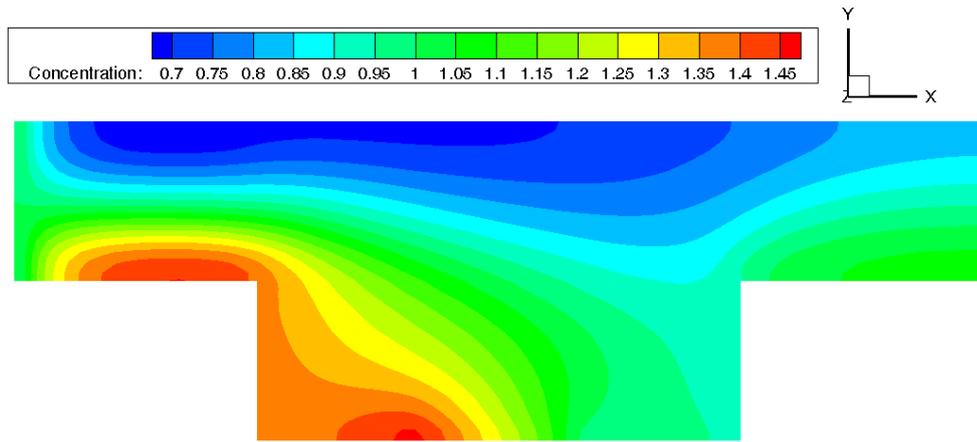


Рисунок 4.14 – Распределение концентрации примеси в центральном сечении

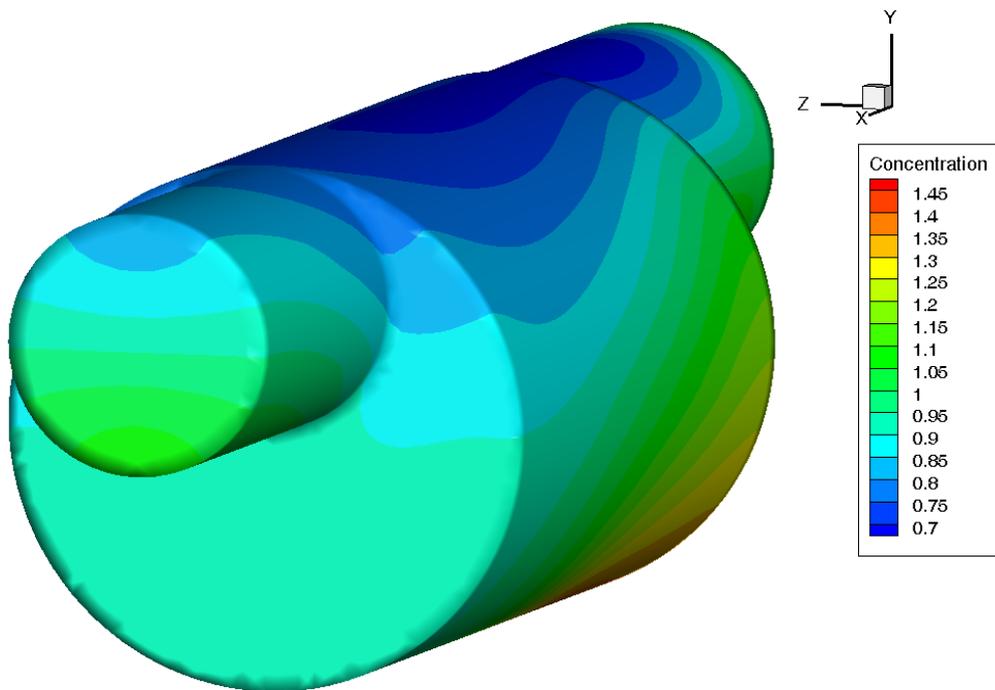


Рисунок 4.15 – Распределение концентрации примеси, общий вид

#### 4.3.2 Задача о сорбционной очистке воздушной среды

Для моделирования очистки гранулированными сорбентами в трехмерном случае рассматривалась расчетная область с размерами  $L_1 = 2, L_2 = 12$ , содержащая 210 сферических гранул радиуса 0.15. На входе фильтра задавалось постоянное распределение скорости  $u_x = 1.0$ . В расчете использовалась тетраэдральная сетка, содержащая 1 780 727 элементов. Расчет проведен на суперкомпьютере К60 (ИПМ им. М.В. Келдыша РАН).

Результирующее стационарное состояние потока при  $Re = 150$  представлено на рисунках 4.16 и 4.17.

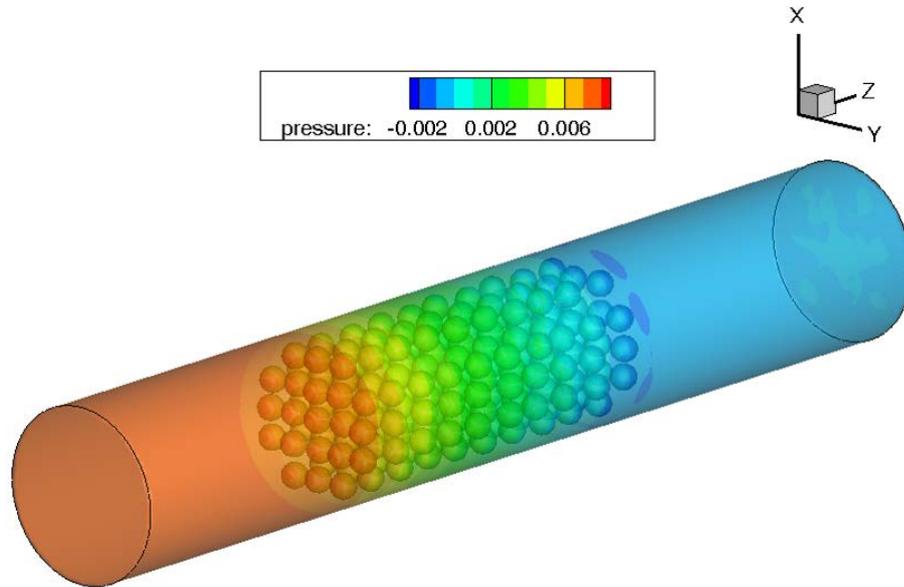


Рисунок 4.16 – Распределение давления

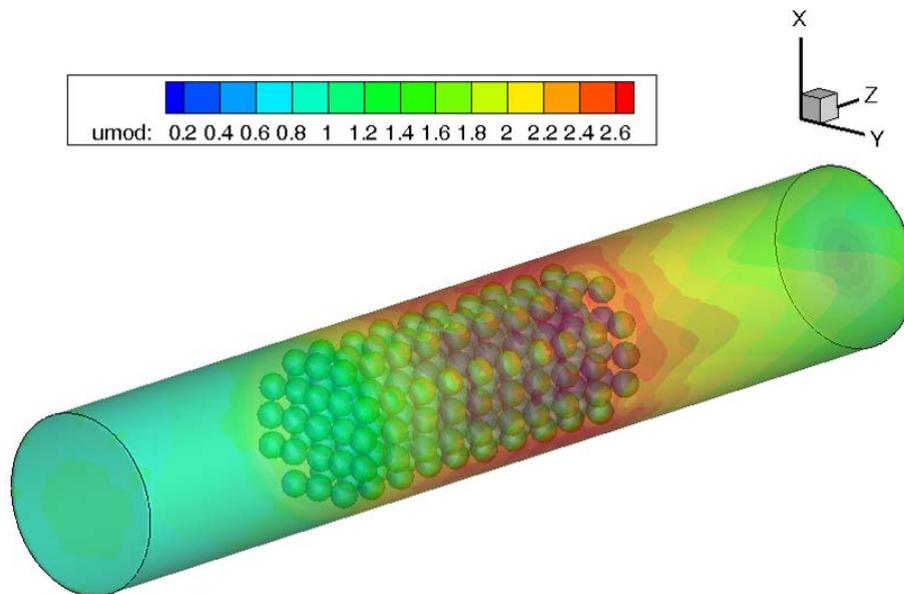


Рисунок 4.17 – Распределение модуля скорости

Моделирование эффектов сорбции проводилось с коэффициентом  $D_0 = 1.0$ , от начального состояния  $C = 0.0$  с равновесной концентрацией примеси равной  $0.0$ . Скорость захвата загрязнителя принималась равной  $A = 1.0$ . Критерий насыщения –  $C^{**} = 1000$ . Результаты проведенных расчетов представлены на рисунках 4.18, 4.19. Они также подтверждают

наличие эффекта очистки – прилипания частиц к поверхности гранул сорбента.

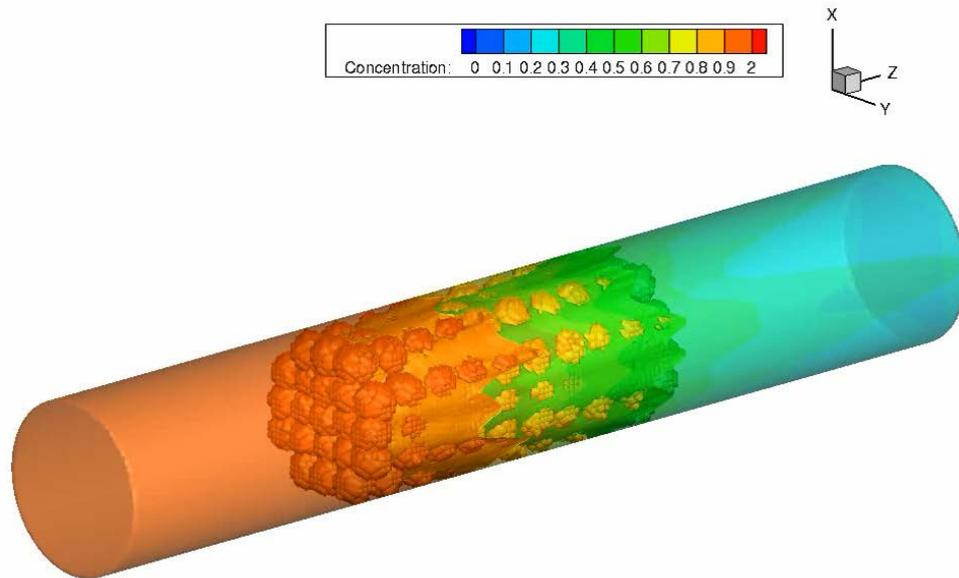


Рисунок 4.18 – Распределение концентрации примеси, изоповерхности

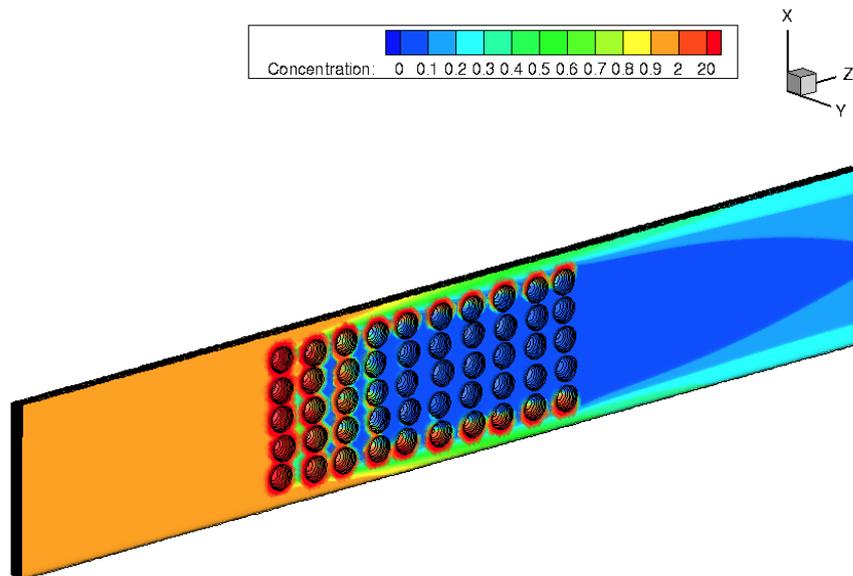


Рисунок 4.19 – Распределение концентрации примеси, набор сечений

### 4.3.3 Задача о загрязнении тепловыделяющего элемента

Расчеты проводились на тетраэдральной сетке, содержащей 140 634 элемента. Распределения параметров установившегося течения при числах Рейнольдса и Прандтля –  $Re = 100$ ,  $Pe = 10$  представлены на рис. 4.20 – 4.22. Число Грасгофа принималось равным  $Gr = 10$ .

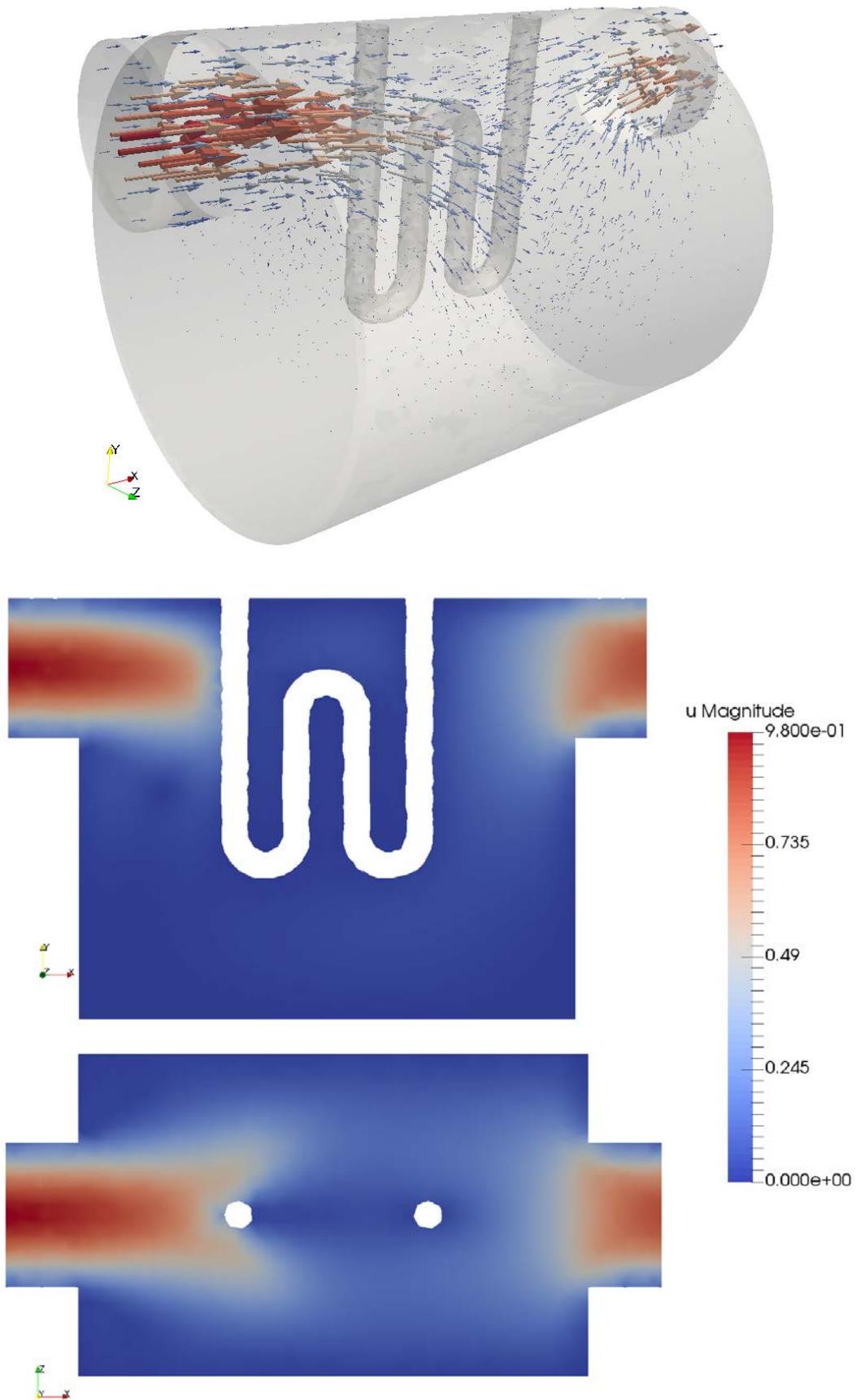


Рисунок 4.20 – Модуль скорости: общий вид (сверху), сечение  $Z=0$  (по центру) и сечение  $Y=0.5$  (снизу)

Рисунок 4.21 – Давление в сечении  $Z=0$ Рисунок 4.22 – Температура в сечении  $Z=0$ 

Результаты моделирования протока жесткой воды, насыщенной  $\text{Ca}(\text{HCO}_3)_2$ , при коэффициенте диффузии  $D=0.005$  для обеих объемных величин –  $\text{CO}_2$  и  $\text{Ca}(\text{HCO}_3)_2$ . Начальное распределение примеси принималось равным нулю. На момент времени  $t=50$ , распределение  $\text{Ca}(\text{HCO}_3)_2$  представлено на рис. 4.23. Концентрация осадка  $\text{CaCO}_3$  на нагревательном элементе изображена на рис. 4.24, а распределение температуры в сечении  $Z=0$  на рис. 4.25. Из сравнения рис. 4.22 и 4.25 видно существенное падение температуры в исследуемой области.

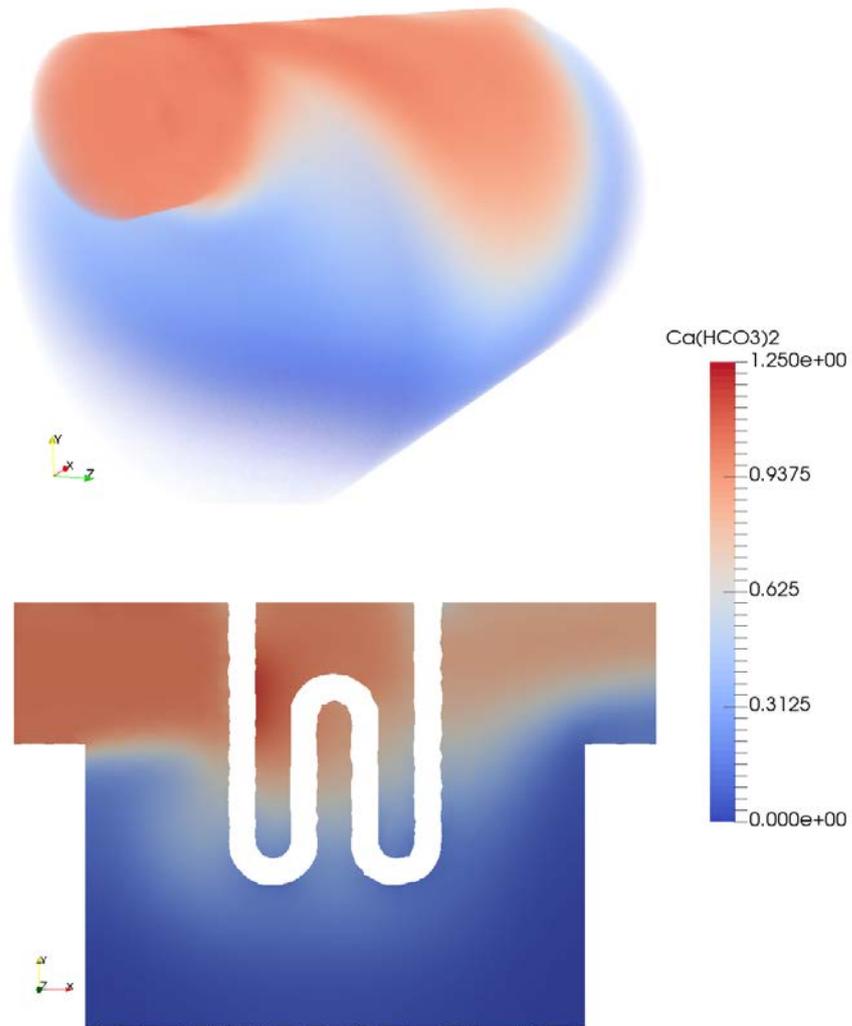


Рисунок 4.23 — Распределение  $\text{Ca}(\text{HCO}_3)_2$  общий вид (сверху) и в сечении  $Z=0$  (снизу)

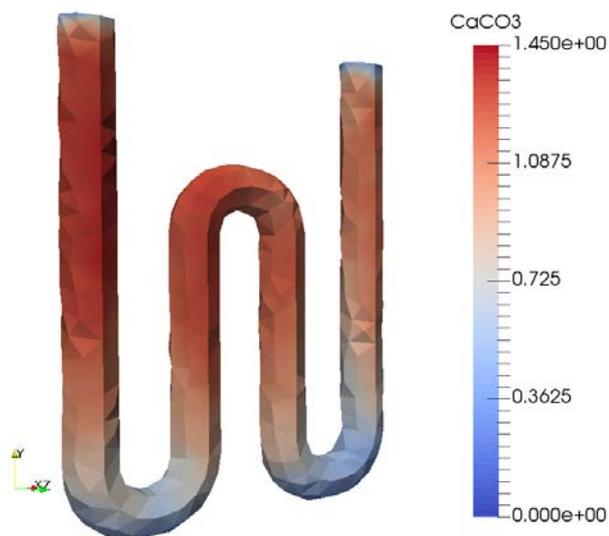


Рисунок 4.24 – Распределение концентрации  $\text{CaCO}_3$  на поверхности нагревательного элемента

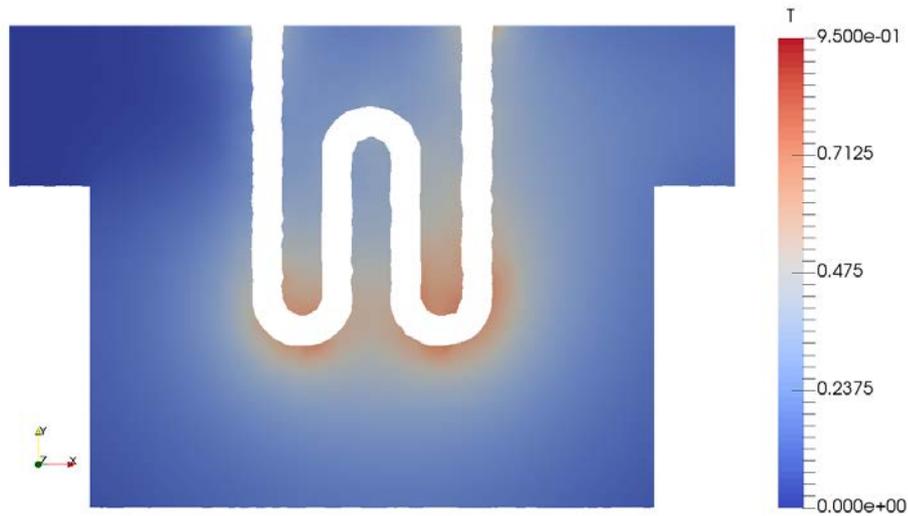


Рисунок 4.25 – Распределение температуры в сечении  $Z=0$

Последующая прокачка раствора соляной кислоты на протяжении  $t = 50$  позволила восстановить функциональность нагревательного элемента и избавиться от большей части осадка. Результаты данного вычислительного эксперимента приведены на рис. 4.26 и 4.27.

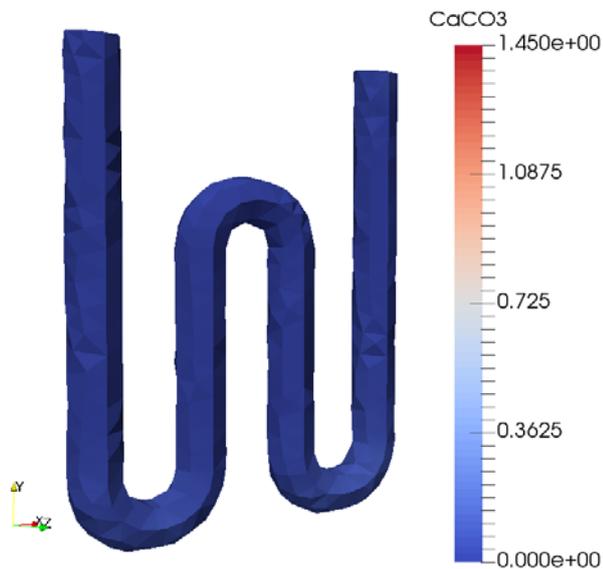


Рисунок 4.26 – Распределение концентрации  $\text{CaCO}_3$  на поверхности нагревательного элемента после регенерации устройства

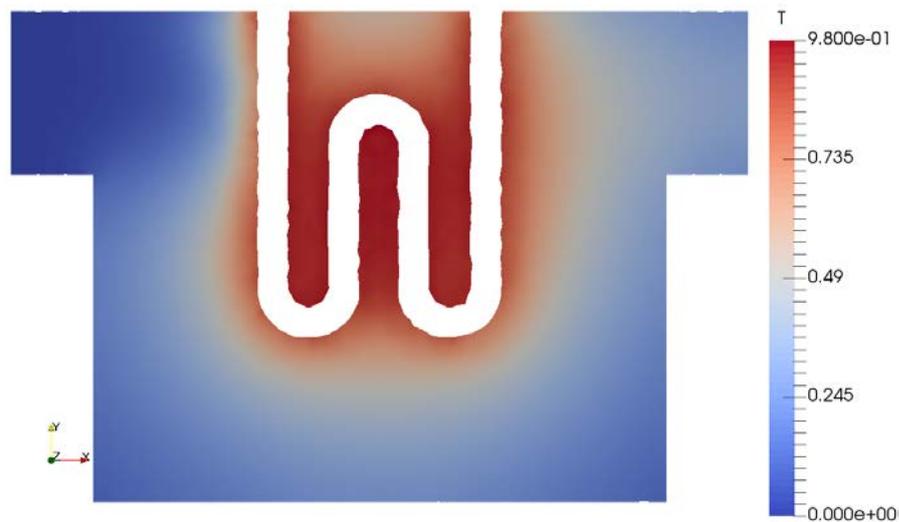


Рисунок 4.27 – Распределение температуры в сечении  $Z=0$  после регенерации устройства

Представленные в п. 4.3 результаты моделирования хорошо соотносятся с теоретическими представлениями, в том числе соответствуют положениям, представленным в работах [14, 15, 18, 74 - 76].

#### 4.6 Выводы по Главе 4

В четвертой главе представлены результаты верификации программной реализации на основе тестовых задач, приведены результаты валидационного комплексного численного расчет об обессоливании воды.

Приведены результаты, полученные при расчете модельных задач фильтрации, постановки которых предложены в первой главе. Проведенные тестовые расчеты показали общую корректность параллельной программной реализации и разработанных численных алгоритмов для решения уравнений моделей сплошной среды при моделировании процессов электромагнитной водной среды и сорбционной очистки воздушного потока. Полученные решения демонстрируют эффект очистки в ожидаемом диапазоне безразмерных параметров. Точность полученных безразмерных количественных характеристик проверена дополнительно при расчетах на вложенных сетках и соответствует порядку аппроксимации пространственных и временных членов уравнений МСС.

Основные результаты данной главы опубликованы в работах [24, 26, 34] в части электромагнитной очистки, [29, 31 – 33] в части фильтрации воздуха и [36] в части образования осадка на нагревательном элементе и регенерации устройства.

## Заключение

В рамках диссертации были рассмотрены применяемые на практике методы тонкой очистки водной и воздушной сред. В качестве целевых способов были выбраны: очистка воды электромагнитным способом и очистка воздуха гранулированными сорбентами. В процессе анализа вышеуказанных процессов был разработан многомасштабный численный подход, сочетающий макроскопические модели механики сплошной среды и модели динамики частиц.

В качестве моделей макроскопического уровня были выбраны квазигидро- и квазигазодинамические модели для описания движущихся водных и/или воздушных сред и система уравнений конвекции-диффузии для описания эволюции загрязняющих потоки примесей. Дополнительно в рамках подхода могут решаться уравнения электро- и магнитостатики, учитывающие прямое влияние электромагнитного поля на заряженные частицы в среде, а также обратное влияние заряженных частиц на конфигурацию поля. В качестве моделей на микроскопическом уровне предложено использовать уравнения динамики частиц, в том числе модели молекулярной динамики.

Комбинирование моделей производится на уровне общего численного алгоритма, базирующегося на методе расщепления по физическим процессам и масштабам. При этом основу численной реализации макроскопических моделей составляет сеточный метод конечных объемов, применяющийся на неструктурированных сетках. Основу численной реализации методов частиц составляет схема Верле и различные модели частиц.

Программное вычислительное ядро, реализующая численные алгоритмы макромасштабного уровня, разработано на языке C++ с использованием библиотек OpenMP и OpenMPI и основано на использовании объектно-ориентированного и программного подходов. Данная программная реализация позволяет динамически комбинировать применяемые модели

посредством конфигурационного файла, поддерживается широкий набор неструктурированных сеток. Предложенная архитектура позволяет использовать единый код, как для двухмерных, так и для трехмерных численных расчетов.

На основе современных веб-технологий разработана цифровая платформа для проведения вычислительных экспериментов на компьютерах и суперкомпьютерах. Основными задачами данной веб-системы является обеспечение графическим интерфейсом классических пакетных приложений с помощью текстового файла в специальном формате, а также унификация взаимодействия пользователей с удаленными вычислительными ресурсами. Представленная платформа позволяет осуществлять подготовку вычислительного эксперимента, запуск заданий на расчет, слежение и управление их прохождением, анализ результатов работы приложения.

Посредством цифровой платформы были проведены верификационные и валидационные расчеты вычислительного ядра, а также серия модельных расчетов, показавшая применимость выбранных численных подходов и алгоритмов для моделирования процессов фильтрации, а также корректность работы вычислительного ядра платформы.

Основными результатами диссертационной работы являются:

- разработаны комплексные математические модели, применяемые в задачах экологической направленности (процессы фильтрации, загрязнение функциональных элементов конструкции), а именно:

а) квазигидродинамическая модель течения водной среды с учетом динамики электромагнитного поля и процессов конвекции и диффузии заряженных примесей;

б) квазигазодинамическая многокомпонентная модель течения воздушной среды с учетом наличия в ней твердых наноразмерных частиц;

в) квазигидродинамическая многокомпонентная модель течения жидкой среды с учетом процессов конвекции, диффузии и химических превращений,

и способы их уточнения и агрегирования в рамках многомасштабного подхода;

- предложены численные методы анализа разработанных комплексных моделей, основанные на методе конечных объемов на неструктурированных сетках, и их программные реализации, ориентированные на использование современных компьютерных и суперкомпьютерных вычислительных систем;

- создана цифровая платформа, включающая вычислительное ядро, базирующееся на разработанных комплексных моделях, и дополнительный программный инструментарий, реализующий всю цепочку вычислительного эксперимента при решении актуальных прикладных задач экологической направленности;

- в целях верификации и валидации комплексных математических моделей и их программной реализации проведено численное моделирование для ряда практических задач промышленной экологии, подтвердившее эффективность разработанной компьютерной технологии.

Перспективы дальнейших исследований заключаются в следующем:

– расширение круга математических моделей для решения актуальных задач промышленной экологии;

– расширение сервисных возможностей вычислительного ядра (поддерживаемые форматы и методы конструирования расчетных сеток, поддержка узловых схем расчета и т.д.), а также реализаций предложенных численных алгоритмов с целью расширения и оптимизации расчетных возможностей вычислительного ядра;

– совершенствование архитектуры и развитие интерфейсов цифровой платформы, в том числе:

- увеличение набора доступных компонент и форм ввода исходных

данных, например, компонент задания геометрии;

- встраивание стандартизированной системы управления версиями для хранения приложений (примером может служить git или svn);
- реализация возможности разделения пользователей, приложений и сценариев по группам для организации интегрированной работы над различными проектами.

### Список литературы

1. Дерябин, В. А. Экология: учебное пособие / В. А. Дерябин, Е. П. Фарафонтова. - Екатеринбург: Изд-во Урал. ун-та, 2016. - 136 с.
2. Сытник, Н.А. Промышленная экология. Учебник для студентов направления подготовки 05.03.06 Экология и природопользование очной и заочной форм обучения / Н.А. Сытник, Назимко. – ФГБОУ ВО «Керченский государственный морской технологический университет». – Керч, 2019. – 134 с.
3. Григорьев, Л.Н., Буренина Т.И. Промышленная экология. Часть 1: Учебное пособие / Л.Н. Григорьев, Т.И. Буренина. – ГОУ ВПО СПбГТУ РП. – СПб., 2004. – 104 с.
4. Храмова, Л.Н. Основы промышленной экологии: учеб. пособие / Л.Н. Храмова, Р.А. Степень, С.В. Соболев. – Красноярск: Сибирский федеральный ун-т, 2012. – 240 с.
5. Промышленная экология: метод. рекомендации по изучению дисциплины и выполнению контрольных работ / Сост. М.М. Морозова. – Ульяновск: УВАУ ГА, 2007. – 58 с.
6. Об охране окружающей среды: федеральный закон от 10.01.2002 № 7-ФЗ; в ред. от 09.03.2021 / Собрание законодательства РФ. – 2002. – № 2. – Ст. 133.
7. Shvets, V. I. Preparation of deuterated inosine suitable for biomedical application / V. I. Shvets, A. M. Yurkevich, O. V. Mosin, D. A. Skladnev // Journal of Medical Sciences. – 1995. – Vol. 8. – № 4. – P. 231-232.
8. Мосин, О. В. Дейтерий, тяжелая вода, эволюция и жизнь / О. В. Мосин // Водоочистка, водоподготовка, водоснабжение. – 2009. – № 8. – С. 64-70.
9. Мосин, О. В. Исследование биосинтеза аминокислот факультативным метилотрофом *Brevibacterium methylicum* на средах,

содержащих тяжелую воду / О. В. Мосин, Д. А. Складнев, Т. А. Егорова, А. М. Юркевич, В. И. Швец // Биотехнология. – 1996. – № 3. – С. 3-12.

10. Szkatula, A. Magnetic treatment of industrial water. Silica activation / A. Szkatula, M. Balanda, M. Kopec // The European Physical Journal Applied Physics. – 2002. – № 18. – P. 41-49.

11. Jia, Zh. Preparation and application of novel magnetically separable  $\gamma$ -Fe<sub>2</sub>O<sub>3</sub>/activated carbon sphere adsorbent / Zh. Jia, K. Peng, Ya. Li, R. Zhu // Materials Science and Engineering: B. – 2011. – № 176 (11) – P. 861-865.

12. Alimi, F. Effect of magnetic water treatment on calcium carbonate precipitation: Influence of the pipe material / F. Alimi, M.M. Tlili, M. Ben Amor, G. Maurin, C. Gabrielli // Chemical Engineering and Processing: Process Intensification. – 2009. – № 48 (8). – P. 1327-1332.

13. Coey, J. M. D. Magnetic Water Treatment / J.M.D Coey, S. Cass // Journal of Magnetism and Magnetic Materials. – 2000. – № 209(1). – P.71-74.

14. Очков, В. Ф. Магнитная обработка воды: история и современное состояние / В. Ф. Очков // Энергосбережение и водоподготовка. – 2006. – № 2. – URL: [http://twt.mpei.ac.ru/ochkov/mo/art\\_ev/](http://twt.mpei.ac.ru/ochkov/mo/art_ev/) (дата обращения: 14.09.2022).

15. Очков, В. Ф. Вода и магнит / В. Ф. Очков // Водоочистка, Водоподготовка, Водоснабжение. – 2011. – № 10. – С. 36-48. – URL: <http://twt.mpei.ac.ru/ochkov/mo/bbb.html> (дата обращения: 14.09.2022).

16. Штокман, Е. А. Очистка воздуха / Е. А. Штокман – Москва: АСВ, 2007. – 313 с.

17. Xu, Zh.: Fundamentals of Air Cleaning Technology and Its Application in Cleanrooms / Zh. Xu – Springer: Berlin, Heidelberg, 2014. – 871 p.

18. Sparks, T. Filters and Filtration Handbook (Sixth Edition) / T. Sparks, G. Chase – Butterworth-Heinemann. – 2015. – 431 p.

19. Лунев, Н. А. Стендовое моделирование газодинамических процессов в воздухоочистительных системах газотурбинных установок / Н. А. Лунев // Фундаментальные исследования. – 2015. – № 5. – С. 104-108.

20. Гладышев, Н. Ф. Системы и средства регенерации и очистки воздуха обитаемых герметичных объектов / Н. Ф. Гладышев, Т. В. Гладышева, С. И. Дворецкий. – Москва: Издательский дом «Спектр», 2016. – 204 с.
21. Мазур, И. И. Курс инженерной экологии: учебник для вузов / И. И. Мазур, О. И. Молдаванов. – 2-е изд., испр. и доп. – Москва: «Высшая школа». – 2001. – 510 с.
22. Понкратова, С. А. Математическое моделирование и управление качеством очистки сточных вод / С. А. Понкратова, В. М. Емельянов, А. С. Сироткин, М. В. Шулаев // Вестник Казанского технологического университета. – 2010. – № 6. – С. 76-85.
23. Пономарев, В. Я. Математическое моделирование процесса аэробной очистки сточных вод предприятий пищевой промышленности / В. Я. Пономарев, Э. Ш. Юнусов, Г. О. Ежкова // Вестник Казанского технологического университета. – 2011. – №17. – С. 139 – 145.
24. Kudryashova, T. A novel parallel algorithm for 3D modelling electromagnetic purification of water / T. Kudryashova, S. Polyakov, N. Tarasov // MATEC Web of Conferences 210. – 04027. - 2018. – 1-5 pp. – URL: <https://doi.org/10.1051/matecconf/201821004027> (дата обращения: 14.09.2022).
25. Тарасов, Н. И. Моделирование потока несжимаемой вязкой жидкости с помощью метода двойного потенциала / Н. И. Тарасов, Ю. Н. Карамзин, Т. А. Кудряшова, С. В. Поляков // Препринты ИПМ им. М. В. Келдыша. – 2018. – № 247. – 20 с. URL: <http://library.keldysh.ru/preprint.asp?id=2018-247> (дата обращения: 14.09.2022)
26. Polyakov, S. Optimization of Parallel Computations for Modeling Water Purification Processes by Electromagnetic Method / S. Polyakov, T. Kudryashova, N. Tarasov // EngOpt 2018 Proceedings of the 6th International Conference on Engineering Optimization. – Springer, Cham. – 2019. – Pp. 754-765.

27. Tatyana Kudryashova, Sergey Polyakov, and Nikita Tarasov. Application of the Double Potential Method to Simulate Incompressible Viscous Flows / T. Kudryashova, S. Polyakov, N. Tarasov // J.M.F. Rodrigues et al. (Eds.): ICCS 2019, LNCS 11539. – 2019. - Pp. 568-579.

28. Тарасов, Н.И. Моделирование потока вязкой несжимаемой жидкости с помощью квазигазодинамической системы уравнений / Н.И. Тарасов, С.В. Поляков, Ю.Н. Карамзин, Т.А. Кудряшова, В.О. Подрыга, Д.В. Пузырьков // Математическое моделирование. – 2019. – т. 31, № 12. – с. 33-43. Перевод: Tarasov N.I. Incompressible Viscous Flow Simulation Using the Quasi-Hydrodynamic Equations' System / N.I. Tarasov, S.V. Polyakov, Yu.N. Karamzin, T.A. Kudryashova, V.O. Podryga, D.V. Puzyrkov // Mathematical Models and Computer Simulations. – 2020. – Vol. 12, No. 4. – Pp. 553-560.

29. Поляков С.В. Многомасштабное суперкомпьютерное моделирование процессов очистки газа методом адсорбции / С.В. Поляков, Ю.Н. Карамзин, Т.А. Кудряшова, В.О. Подрыга, Д.В. Пузырьков, Н.И. Тарасов // Вычислительные методы и программирование. – 2020. – Т. 21. – С. 64-77.

30. Поляков С.В. Метод двойного потенциала для моделирования внутреннего течения вязкой несжимаемой жидкости / С.В. Поляков, Т.А. Кудряшова, Н.И. Тарасов // Доклады РАН. Математика, информатика, процессы управления. – 2020. – Т. 494, № 1. – С. 73-76. Перевод: Polyakov S.V. Double potential method for modeling the internal flow of a viscous incompressible liquid / S.V. Polyakov, T.A. Kudryashova, N.I. Tarasov // Doklady Mathematics. – 2020. – Vol. 102, No. 2. – Pp. 418-421.

31. Podryga, V.O. Developing of Multiscale Approach to HPC-Simulation of Multiphase Fluid Flows / V.O. Podryga, S.V. Polyakov, N.I. Tarasov // Lobachevskii Journal of Mathematics. – 2021. – Vol. 42, No. 11. – Pp. 2623-2633.

32. Поляков, С.В. Применение многомасштабного подхода для моделирования сорбционной фильтрации воздуха / С.В. Поляков, Т.А.

Кудряшова, Н.И. Тарасов // Доклады РАН. Математика, информатика, процессы управления. – 2021. – том 500. - С. 1-5. Перевод: Polyakov S.V. Application of the Multiscale Approach to Simulation of Air Sorbent Filtration / S.V. Polyakov, T.A. Kudryashova, N.I. Tarasov // Doklady Mathematics. – 2021. – Vol. 104, No. 5. – Pp. 297-300.

33. Tarasov, N. Computer modeling of air flows purification using sorption filters / N. Tarasov, Yu. Karamzin, T. Kudryashova, V. Podryga, S. Polyakov // Journal of Physics: Conf. Series (JPCS). – 2021. – V. 2028. – Paper 012025. – 7 p.

34. Kudryashova, T.A. Mathematical Modelling of Electrophysical Water Treatment / T.A. Kudryashova, S.V. Polyakov, N.I. Tarasov // Defect and Diffusion Forum. – 2021. – vol. 412. – Pp. 149-162.

35. Тарасов, Н.И. Архитектура и реализация цифровой платформы для проведения вычислительных экспериментов на суперкомпьютерах / Н.И. Тарасов // Препринты ИПМ им. М.В. Келдыша. 2022. № 50. 30 с.

36. Тарасов, Н.И. Моделирование образования и удаления известкового налета в системах водоочистки / Н.И. Тарасов, Т.А. Кудряшова, С.В. Поляков // Доклады Российской Академии Наук. Математика, информатика, процессы управления. – 2022. – Том 505. – с. 79-85. Перевод: Tarasov, N.I. Modeling Formation and Removal of Limescale in Water Treatment Systems / N.I. Tarasov, T.A. Kudryashova, S.V. Polyakov // Doklady Mathematics. – 2022. – Vol. 106, No. 1. – P. 279-285.

37. Роуч, П. Вычислительная гидродинамика / П. Роуч. – Москва: Мир, 1980. – 618 с.

38. Флетчер, К. Вычислительные методы в динамике жидкостей: В 2-х т. Т.2.: Пер. с англ. / К. Флетчер. – Москва: Мир, 1991. – 552 с.

39. Richardson, S.M. Solution of three-dimensional incompressible flow problems / S.M. Richardson, A.R. Cornish // J. Fluid Mech. – 1977. – Vol. 82 – Is. 2 – P. 309-319.

40. Richardson, S.M. Numerical solution of the three—dimensional Navier—Stokes equations. Doctoral dissertation / S.M. Richardson – London: Department of Chemical Engineering and Chemical Technology: Imperial College of Science and Technology. – 1976. – 218 p.
41. Gegg, S.G. A dual potential formulation of the Navier-Stokes equations. Doctoral dissertation / S.G. Gegg – Iowa: Iowa State University. – 1989. – 201 p.
42. Chetverushkin, B.N. Kinetic schemes and quasi-gasdynamics system of equations / B.N. Chetverushkin – Barcelona: CIMNE. – 2008. – 298 p.
43. Elizarova, T.G. Quasi-gas dynamic equations / T.G. Elizarova – Berlin, Heidelberg, NY: Springer-Verlag. – 2009 – 286 p.
44. Шеретов, Ю.В. Динамика сплошных сред при пространственно-временном осреднении / Ю.В. Шеретов – М.-Ижевск: НИЦ "Регулярная и хаотическая динамика" – 2009 – 400 с.
45. Злотник, А.А. О дискретизации одномерной квазигидродинамической системы уравнений для реального газа / А.А. Злотник, В.А. Гаврилин // Вестник Московского энергетического института. – 2016. – № 1. – С. 5-14.
46. Елизарова, Т.Г. О квазигазо- и гидродинамических уравнениях бинарных смесей газов / Т.Г. Елизарова, А.А. Злотник, Б.Н. Четверушкин // ДАН. – 2014. – Том 459, №4. – С.395-399.
47. Злотник, А.А. Энтропийно консервативная пространственная дискретизация многомерной квазигазодинамической системы уравнений / А.А. Злотник // Ж. вычисл. матем. и матем. физ. – 2017. – Том 57, №4. - С. 710-729.
48. Кудряшова Т.А., Подрыга В.О., Поляков С.В. Параллельная программа расчета на ЦПУ течений газа в микроканалах на основе уравнений квазигазодинамики GIMM\_APP\_QGD\_CPU программного комплекса GIMM\_NANO, версия 1. Свидетельство о государственной

регистрации программы для ЭВМ № 2016612717 от 09.03.2016.  
Правообладатель: ИПМ им. М.В.Келдыша РАН

49. Кудряшова Т.А., Подрыга В.О., Поляков С.В. Параллельная программа расчета на ЦПУ течений газа в микроканалах на основе уравнений квазигазодинамики и методов молекулярной динамики GIMM\_APP\_QGD\_MD\_CPU программного комплекса GIMM\_NANO, версия 2. Свидетельство о государственной регистрации программ для ЭВМ № 2018666727, 20 декабря 2018. Правообладатель: ИПМ им. М.В.Келдыша РАН.

50. Атмосфера стандартная. Параметры: ГОСТ 4401-81: Межгосударственный стандарт: утвержден и введен в действие Постановлением Государственного комитета СССР по стандартам от 27.02.81 № 145: взамен ГОСТ 4401-73: издание с Изменением № 1, утвержденным в феврале 1987 года: дата введения 01.05.2004 – ИПК Издательство стандартов, 2004. – 180 с.

51. Сычев, В.В. Термодинамические свойства воздуха / В.В. Сычев, А.А. Вассерман, А.Д. Козлов, Г.А. Спиридонов, В.А. Цымарный — ГСССД. Серия монографии. М.: Издательство стандартов, 1978. – 276 с.

52. Варгафтик, Н.Б. Справочник по теплофизическим свойствам газов и жидкостей / Н.Б. Варгафтик – М., 1972 г. – 720 с.

53. Подрыга В.О., Поляков С.В. Параллельная программа расчета на ЦПУ термодинамического равновесия в микросистемах газ-металл GIMM\_APP\_MD\_CPU\_Gas\_Metal программного комплекса GIMM\_NANO, версия 1. Свидетельство о государственной регистрации программы для ЭВМ № 2015618870 от 19.08.2015. Правообладатель: ИПМ им. М.В.Келдыша РАН.

54. Пузырьков Д.В., Поляков С.В., Подрыга В.О. Программный комплекс для решения задач распределенной генерации больших молекулярных систем KIAM\_MolSDAG\_CPU, версия 1. Свидетельство о

государственной регистрации программ для ЭВМ № 2020618460, 29 июля 2020. Правообладатель: ИПМ им. М.В.Келдыша РАН.

55. Rapaport, D.C. The Art of Molecular Dynamics Simulation / D.C. Rapaport – Cambridge: Cambridge University Press, 2004. – 564 p.

56. Verlet, L. Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules / L. Verlet // Phys. Rev., 1967. – Vol. 159. – Pp. 98-103.

57. Podryga, V. O. Multiscale approach to computation of three-dimensional gas mixture flows in engineering microchannels / V. O. Podryga // Dokl. Math. – 2016. – № 94. – P. 458-460.

58. Подрыга, В. О. Многомасштабное численное моделирование течений газа в каналах технических микросистем: дис. на соискание степени д-ра физ.-мат. наук: 05.13.18 / Подрыга Виктория Олеговна. – Москва, 2017. – 229 с.

59. Подрыга, В. О. Определение макропараметров реального газа методами молекулярной динамики / В. О. Подрыга // Матем. моделирование – 2015. – Том 7. – № 27 – С. 80-90.

60. Подрыга, В. О. Молекулярно-динамический расчет коэффициента диффузии газов на примере аргона, азота, водорода, кислорода, метана и углекислого газа / В. О. Подрыга, Е. В. Вихров, С. В. Поляков // Препринты ИПМ им. М.В. Келдыша. – 2019. – № 96. – 24 с.

61. Подрыга, В.О. Многомасштабное моделирование истечения газовой струи в вакуум / В.О. Подрыга, С.В. Поляков // Препринты ИПМ им.М.В.Келдыша. 2016. № 81. 52 с.

62. Подрыга, В.О. Суперкомпьютерное многомасштабное моделирование течений газовых смесей в микроканалах / В.О. Подрыга, С.В. Поляков // Вычислительные методы и программирование, 2018. – Том 19, № 1. – С. 38-50.

63. Lennard–Jones, J.E. Cohesion / J.E. Lennard-Jones // Proceedings of the Physical Society. – 1931. – Vol. 43, No. 5. – P. 461–482.

64. Morse, T.F. Kinetic model equation for gas mixture / T.F. Morse // *Phys. Fluids.* – 1964. – Vol. 7(12). – P. 2012–2013.
65. Daw, M.S. Model of metallic cohesion: The embedded–atom method / M.S. Daw // *Phys. Rev.* – 1989. – Vol. 39, No. 11. – P. 7441–7452.
66. Подрыга, В.О. Молекулярно-динамический расчет макропараметров газа в потоке и на границе. / В.О. Подрыга, С.В. Поляков // *Препринты ИПМ им. М.В. Келдыша*, 2016. – № 80. – 26 с.
67. Подрыга, В.О. Молекулярно-динамический расчет макропараметров технических газов на примере аргона, азота, водорода и метана / В.О. Подрыга, Е.В. Вихров, С.В. Поляков // *Математическое моделирование.* – 2019. – Том 31, № 8 – С. 44-60.
68. Подрыга, В.О. Молекулярно-динамический расчет коэффициента диффузии газов на примере аргона, азота, водорода, кислорода, метана и углекислого газа / В.О. Подрыга, Е.В. Вихров, С.В. Поляков // *Препринты ИПМ им. М.В. Келдыша*, 2019. – № 96. – 24 с.
69. Иванов, М.Г. Размерность и подобие / М.Г. Иванов – *Долгопрудный*, 2013. 68 с.
70. Зуев, К.И. Основы теории подобия: конспект лекций / К. И. Зуев. – *Владим. гос. ун-т.– Владимир: Изд-во Владим. гос. ун-та*, 2011. – 51 с.
71. Борзенко, Е. И. Исследование явления проскальзывания в случае течения вязкой жидкости в изогнутом канале / Е. И. Борзенко, О. А. Дьякова, Г. Р. Шрагер // *Вестн. Томск. гос. ун-та. Матем. и мех.* – 2014. – № 2(28). – С. 35–44.
72. Захаренков, М. Н. Реализация граничных условий частичного или полного проскальзывания при решении уравнений Навье–Стокса в переменных функция тока–завихренность / М. Н. Захаренков // *Ж. вычисл. матем. и матем. физ.* – 2001. – Том 41, № 5. – С. 796–806.
73. Sankaranarayanan, S. Three-dimensional finite difference model for transport of conservative pollutants / S. Sankaranarayanan, N.J. Shankar, H.F. Cheong // *Ocean Engineering.* – 1998. – Vol. 25, Issue 6. – P. 425-442.

74. Crooks, R.M. Electrochemically mediated desalination / R. M. Crooks, K. N. Knust, M. R. Stanley, F. J. Carrillo, D. Hlushkou, U. Tallarek / Proc. of 18th International Conference on Miniaturized Systems for Chemistry and Life Sciences. – 2014. – P. 156-159.

75. Knust, K.N. Electrochemically Mediated Seawater Desalination / K.N. Knust, D. Hlushkou, R.K. Anand, U. Tallarek, R.M. Crooks // Angew. Chem. Int. Ed. – 2013 – Vol. 52, Issue 31 – P. 8107-8110.

76. Influence of Water Scale on Thermal Flow Losses of Domestic Appliances / D. Dobersek, D. Goricanec // Int. J. Math. Models Methods Appl. Sci. - 2007. – Vol. 1, Issue 2. – P. 55-61.

77. Самарский, А.А. Разностные схемы на нерегулярных сетках / А.А. Самарский, А.В. Колдоба, Ю.А. Повещенко, В.Ф. Тишкин, А.П. Фаворский – Минск, ЗАО «Критерий», 1996. – 196 с.

78. Ильин, В. П. Методы конечных разностей и конечных объемов для эллиптических уравнений / В. П. Ильин – Новосибирск, Изд-во Ин-та математики СО РАН, 2000. – 345 с.

79. Eymard, R. The finite volume method / R. Eymard, T. R. Gallouet, R. Herbin // Handbook of Numerical Analysis, Elsevier. – 2000 – Vol. 7 – P. 713-1020.

80. LeVeque, R.J. Finite Volume Methods for Hyperbolic Problems / R.J. LeVeque – Cambridge University Press, 2002. – 558 p.

81. Александров, П.А. О возможности построения консервативного вычислительного метода решения задачи Коши для гамильтоновых систем на основе двухстадийных симметрично-симплектических методов Рунге-Кутты / П.А. Александров, Г.Г. Еленин // Матем. моделирование. – 2014. – Том 26, № 10. – С. 47-63.

82. Иванов, Е. Г. Автоматическая параллельная генерация неструктурированных расчетных сеток для задач вычислительной механики: дис. на соискание степени канд. физ.-мат. наук: 05.13.18 / Иванов Евгений Геннадьевич – Новосибирск, 2007. – 115 с.

83. Головченко, Е. Н. Декомпозиция расчетных сеток для решения задач механики сплошных сред на высокопроизводительных вычислительных системах: дис. на соискание степени канд. физ.-мат. наук: 05.13.18 / Головченко Евдокия Николаевна – Москва, 2014. – 165 с.
84. Подрыга, В. О. Параллельная реализация многомасштабного подхода для расчета микротечений газа / В. О. Подрыга, С. В. Поляков. // Выч. мет. программирование – 2016. – Т. 17 – В. 2 – С. 147-165.
85. Страуструп, Б. Язык программирования C++. Специальное издание. Пер. с англ. / Б. Страуструп – Москва: Издательство Бином, 2011. – 1136 с.
86. Open MPI: Open Source High Performance Computing [сайт]. – URL: <https://www.open-mpi.org/> (дата обращения: 14.09.2022).
87. Specifications – OpenMP [сайт]. – URL: <https://www.openmp.org/specifications/> (дата обращения: 14.09.2022).
88. CUDA Toolkit Documentation [сайт]. – URL: <https://docs.nvidia.com/cuda/> (дата обращения: 14.09.2022).
89. JSON [сайт] – URL: <https://www.json.org/json-en.html> (дата обращения: 14.09.2022).
90. GitHub - nlohmann/json: JSON for Modern C++ [сайт]. - URL: <https://github.com/nlohmann/json> (дата обращения: 14.09.2022).
91. Гасилов, В. А. Пакет прикладных программ MARPLE3D для моделирования на высокопроизводительных ЭВМ импульсной магнитоускоренной плазмы / В. А. Гасилов, С. В. Дьяченко, А. С. Болдарев, О. Г. Ольховская, Е. Л. Карташева, Г. А. Багдасаров // Препринты ИПИМ им. М. В. Келдыша, 2011. – № 20. – 36 с.
92. Kartasheva, E. An Implicit Complexes Framework for Heterogeneous Objects Modelling, in Heterogeneous Objects Modelling and Applications / E. Kartasheva, V. Adzhiev, P. Comminos, O. Fryazinov, A. Pasko // Lecture Notes in Computer Science, Springer-Verlag. – 2008. – Vol. 4889. – P. 1-41.

93. VTK - The Visualization Toolkit [сайт]. – URL: <https://vtk.org/> (дата обращения: 14.09.2022).
94. Поляков, С.В. Гибридная суперкомпьютерная платформа и разработка приложений для решения задач механики сплошной среды сеточными методами / С.В. Поляков, Ю.Н. Карамзин, О.А. Косолапов, Т.А. Кудряшова, С.А. Суков // Известия ЮФУ. Технические науки. – 2012. – №6 – С. 105-115.
95. ЦКП ИПМ РАН [сайт]. – URL: <https://ckp.kiam.ru/?home> (дата обращения: 14.09.2022).
96. Руководство пользователя системы МВС-1000/К-60 / А.В. Баранов, А.О. Лацис, С.В. Сажин, М.Ю. Храмцов. – URL: <https://www.kiam.ru/MVS/documents/k60/userguide.html> (дата обращения: 14.09.2022).
97. Slurm Workload Manager - Documentation [сайт]. – URL: <https://slurm.schedmd.com/documentation.html> (дата обращения: 14.09.2022).
98. Home – Docker [сайт]. – URL: <https://www.docker.com/> (дата обращения: 14.09.2022).
99. Kubernetes Production-Grade Container Orchestration [сайт]. – Режим доступа: <https://kubernetes.io/> (дата обращения: 14.09.2022).
100. Службы облачных вычислений | Microsoft Azure [сайт]. – URL: <https://azure.microsoft.com/ru-ru/> (дата обращения: 14.09.2022).
101. Сервисы облачных вычислений – Amazon Web Services (AWS) [сайт]. – URL: <https://aws.amazon.com/ru/> (дата обращения: 14.09.2022).
102. Надежное облако для вашего бизнеса – Yandex Cloud [сайт]. – URL: <https://cloud.yandex.ru/> (дата обращения: 14.09.2022).
103. Поляков, С.В. Облачный сервис для решения перспективных задач нанотехнологии / С.В. Поляков, В.О. Подрыга, Д.В. Пузырьков // Научный сервис в сети Интернет: труды XIX Всероссийской научной конференции (18-23 сентября 2017 г., г. Новороссийск). – М.: ИПМ им.

М. В. Келдыша, 2017. – С. 379-388. – URL: <http://keldysh.ru/abrau/2017/54.pdf> (дата обращения: 14.09.2022).

104. Пузырьков, Д. В. Облачный сервис для масштабных молекулярно-динамических расчетов: от идеи до реализации / Д. В. Пузырьков, В. О. Подрыга, С. В. Поляков // Научный сервис в сети Интернет: труды XIX Всероссийской научной конференции (18-23 сентября 2017 г., г. Новороссийск). – М.: ИПМ им. М.В.Келдыша, 2017. – С. 406-416. – Режим доступа: <http://keldysh.ru/abrau/2017/58.pdf> (дата обращения: 14.09.2022).

105. Sim Streamlined For Design: structural simulation and analysis in the cloud [сайт]. – URL: <https://www.sim4design.com/index> (дата обращения: 14.09.2022).

106. Nucleonica [сайт]. – URL: <https://www.nucleonica.com/> (дата обращения: 14.09.2022).

107. Everest [сайт]. – URL: <http://everest.distcomp.org/> (дата обращения: 14.09.2022).

108. Sukhoroslov, O. A Web-Based Platform for Publication and Distributed Execution of Computing Applications / O. Sukhoroslov, S. Volkov, A. Afanasiev // 14th International Symposium on Parallel and Distributed Computing (ISPDC). IEEE. – 2015. – P. 175-184.

109. Puzyrkov, D.V. Cloud service for HPC management: ideas and appliance / D.V. Puzyrkov, V.O. Podryga, S.V. Polyakov // Lobachevskii J. Math. – 2018. – Vol. 39, Issue 9 – P. 1251–1261.

110. Пузырьков Д.В., Поляков С.В., Тарасов Н.И. Программная среда для решения задач управления численными расчетами на суперкомпьютерах KIAM\_WMCS, версия 1. Свидетельство о государственной регистрации программ для ЭВМ № 2018666098, 12 декабря 2018. Правообладатель: ИПМ им. М.В.Келдыша РАН.

111. Поляков С.В., Подрыга В.О., Пузырьков Д.В., Тарасов Н.И. Вэб-интерфейс пользователя для моделирования на суперкомпьютерах свойств молекулярных систем KIAM\_MMD\_WUI, версия 1. Свидетельство о

государственной регистрации программ для ЭВМ № 2020661866, 01 октября 2020. Правообладатель: ИПМ им. М.В.Келдыша РАН.

112. The web framework for perfectionists with deadlines | Django [сайт]. – URL: <https://www.djangoproject.com/> (дата обращения: 14.09.2022).

113. Vue.js - The Progressive JavaScript Framework | Vue.js [сайт]. – URL: <https://vuejs.org/> (дата обращения: 14.09.2022).

114. Node.js [сайт]. – URL: <https://nodejs.org/en/> (дата обращения: 14.09.2022).

115. Express - Node.js web application framework [сайт]. – URL: <https://expressjs.com/> (дата обращения: 14.09.2022).

116. Passport.js [сайт]. – URL: <https://www.passportjs.org/> (дата обращения: 14.09.2022).

117. Sequelize | Feature-rich ORM for modern TypeScript & JavaScript [сайт]. – URL: <https://sequelize.org/> (дата обращения: 14.09.2022).

118. SQLite Home Page [сайт]. – URL: <https://sqlite.org/index.html> (дата обращения: 14.09.2022).

119. The WebSocket Protocol / I. Fette, A. Melnikov. - Internet Engineering Task Force, 2011. – RFC 6455 - URL: <https://datatracker.ietf.org/doc/html/rfc6455> (дата обращения: 14.09.2022).

120. ws: a Node.js WebSocket library [сайт]. – URL: <https://github.com/websockets/ws> (дата обращения: 14.09.2022).

121. nginx [сайт]. – URL: <https://nginx.org/> (дата обращения: 14.09.2022).

122. Quasar Framework [сайт]. – URL: <https://quasar.dev/> (дата обращения: 14.09.2022).

123. Home | Vue Router [сайт]. – URL: <https://router.vuejs.org/> (дата обращения: 14.09.2022).

124. What is Vuex? | Vuex [сайт]. – URL: <https://vuex.vuejs.org/> (дата обращения: 14.09.2022).

125. ParaViewWeb [сайт]. – URL: <https://kitware.github.io/paraviewweb/index.html> (дата обращения: 14.09.2022).
126. Visualizer [сайт]. – URL: <https://kitware.github.io/visualizer/> (дата обращения: 14.09.2022).
127. ParaView [сайт]. – URL: <https://www.paraview.org/> (дата обращения: 14.09.2022).
128. Markdown Guide [сайт]. – URL: <https://www.markdownguide.org/> (дата обращения: 14.09.2022).
129. The Official YAML Web Site [сайт]. – URL: <https://yaml.org/> (дата обращения: 14.09.2022).