УДК 004.512 doi: 10.15827/0236-235X.151.468-474 2025. Т. 38. № 3. С. 468–474

Реализация проекта для решения проблемы отсутствия графической оболочки пакета OpenFOAM

Д.И. Читалов [1]✉

[1] Южно-Уральский научный центр, г. Миасс, 456317, Россия

Ссылка для цитирования
Читалов Д.И. Реализация проекта для решения проблемы отсутствия графической оболочки пакета OpenFOAM // Программные продукты и системы. 2025. Т. 38. № 3. С. 468–474. doi: 10.15827/0236-235X.151.468-474
Информация о статье
Группа специальностей ВАК: 1.2.2
Поступила в редакцию: 11.12.2024 После доработки: 10.01.2025 Принята к публикации: 11.02.2025

Аннотация. Отсутствие графического интерфейса в базовом дистрибутиве пакета OpenFOAM значительно усложняет работу пользователей. Данное исследование направлено на устранение этой проблемы. Разработан и интегрирован модуль, предназначенный для проведения численных экспериментов с помощью программы-решателя mulesQHDFoam. Его преимуществами в сравнении с приложениями-аналогами являются открытость исходного кода, а также наличие русифицированного интерфейса. Отделение программного кода, обеспечивающего функционирование фронтенда, от программного кода бэкенда упрощает последующую поддержку и доработку модуля. В работе описаны некоторые аспекты применения решателя mulesQHDFoam, особенности постановки экспериментов с его участием. Представлен перечень средств разработки, в том числе основного языка программирования, дополнительных библиотек и вспомогательных технологий. Процессная диаграмма, созданная с помощью языка построения диаграмм UML, демонстрирует алгоритм работы пользователя с программой. Возможности модуля протестированы в рамках проведения эксперимента по моделированию одной из фундаментальных задач механики сплошных сред. Представлено изображение результатов эксперимента для выбранной задачи с визуализацией главного окна графической оболочки. Благодаря размещению исходного кода на сервисе GitHub программа доступна для применения сторонними специалистами. Проанализирована эффективность выбранного перечня технологий разработки по итогам выполненного исследования, определена целесообразность его дальнейшего применения. Сформулированы краткие результаты работы с указанием потенциальной практической ценности проекта. Приведены основные пункты, отличающие разработанные модули от программных продуктов-аналогов.
Ключевые слова: численное моделирование, OpenFOAM, решатель mulesQHDFoam, открытое ПО, графический интерфейс пользователя, Python, механика сплошных сред

Введение. При проектировании продукции современного высокотехнологичного машиностроения активно применяются пакеты вычислительной гидродинамики, представляющие собой мощный и развитый инструментарий для моделирования широкого спектра физических задач. Они используются международными инженерными компаниями для замены дорогостоящих и затратных по времени этапов разработки продукции численными экспериментами и получением цифровых моделей. Данные программные решения, как правило, распространяются на коммерческой основе, их исходный код закрыт для конечного пользователя, что не позволяет модифицировать существующий функционал, а также добавлять новые опции.

Одним из немногих доступных открытых программных средств для моделирования физических процессов, связанных с продукцией высокотехнологичного машиностроения, является прикладной пакет OpenFOAM. В перечень его возможностей входит работа с большим списком задач *механики сплошных сред* (МСС). Он показал высокую эффективность при построении численных моделей различного уровня сложности. Открытый исходный код позволяет вносить изменения в дистрибутив, расширяя доступ пользователя к новым решателям, а также интегрировать собственные программные модули и решения, что делает OpenFOAM одним из лучших средств для инженерных и научных исследований.

Физически OpenFOAM представляет собой набор библиотек, созданных на языке C++ и предоставляющих пользователю инструментарий для решения систем дифференциальных уравнений в частных производных методом конечных объемов. Эти библиотеки управляют процессом моделирования задач МСС на базе многочисленных программ-решателей, а также работой вспомогательных утилит. Для пользователей работа с OpenFOAM, помимо знания предметной области (гидродинамики, аэродинамики, движения деформируемых твердых тел), требует понимания в отношении подготовки расчетного случая, создания и заполнения файлов-словарей с расчетными параметрами, а также в части знания многих решателей и служебных утилит. Процесс моделирования осложняется тем, что вся работа осуществляется средствами командной строки. В эпоху классических оконных интерфейсов это выглядит неоптимальным и повышает вероятность ошибок на различных этапах моделирования.

Рядом компаний-разработчиков ПО приложены усилия по устранению указанной проблемы и предоставлению инженерам и исследователям более привычных графических интерфейсов, интегрируемых с OpenFOAM. К таким продуктам относятся, например, графические оболочки Helyx-OS, Visual-CFD и enGrid. Они показали свою результативность в связке с OpenFOAM, но разработаны иностранными компаниями, имеют англоязычный интерфейс и в ряде случаев требуют оплаты услуг технической поддержки или приобретения лицензии. Кроме того, из-за наметившегося в России курса на импортозамещение ПО создание собственной графической оболочки, устраняющей указанные недостатки, выглядит логичным и необходимым. Отечественные частные коллективы исследователей проявляют интерес к проблеме создания графических оболочек с открытым исходным кодом [1–3], но в процессе анализа публикационной активности авторов и российского рынка ПО не было обнаружено законченных проектов и полноценных результатов. Зарубежные коллективы исследователей работают над несколькими проектами открытых графических оболочек для OpenFOAM, в частности, над Blender и CfdOF (https://github.com/jaheyns/CfdOF/tree/master/CfdOF) на базе языка программирования Python. Оболочка CfdOF с 2017 г. находится в статусе активной разработки, расширяется новыми утилитами для управления численным моделированием большего перечня задач МСС. Среди российских исследователей данная оболочка не получила распространения, поскольку покрывает не весь перечень задач, моделируемых в OpenFOAM. Кроме того, взаимодействие с ней осложняется англоязычным интерфейсом. Зарубежные исследователи отмечают сложность установки, низкую производительность, высокое потребление оперативной памяти, недостаточный функционал для покрытия всей предметной области, невысокое качество расчетных сеток [4, 5].

С 2015 г. ведется работа над проектом сторонней, но интегрируемой с OpenFOAM русифицированной графической оболочки для управления численным моделированием задач МСС, включая все этапы от подготовки расчетного случая до валидации параметров задачи МСС и запуска процесса вычислений на базе одной из программ-решателей [6, 7]. За это время исходный код продукта существенно расширился и продолжает обновляться в связи с изменениями в дистрибутиве OpenFOAM. Разработчики платформы совершенствуют расчетные алгоритмы, а также меняют подход к структуре расчетных случаев и организации вычислений, что требует поддержания графической оболочки в актуальном состоянии.

Настоящая статья предполагает исследование возможности расширения исходного кода дистрибутива OpenFOAM с помощью внедрения программного средства для управления численными экспериментами на базе решателя mulesQHDFoam. Концепция проекта опирается на многослойную архитектуру базовой версии графической оболочки, что позволяет подключать к ней дополнительные компоненты.

Метод исследования

Рассмотрим проблему на примере разработки программного средства для управления экспериментами по расчету вязкой несжимаемой изотермической жидкости в круглой трубе. Эта проблема МСС относится к области гидродинамики и моделируется на базе решателя mulesQHDFoam. Данная задача применяется для демонстрации точности численных алгоритмов на примере ламинарных течений, а также демонстрации возможностей алгоритмов для моделирования турбулентных течений и режимов ламинарно-турбулентного перехода [8].

В основе проводимого численного моделирования лежит метод с участием регуляризованной системы уравнений Навье – Стокса. Регуляризация уравнений предполагает, что массовая плотность потока жидкости отличается от среднего импульса единицы объема на малую величину. Такое отличие приводит к появлению малых диссипативных добавок в исходной системе уравнений, которые обеспечивают невозрастание кинетической энергии. Применительно к вычислительному алгоритму эти слагаемые позволяют использовать условно устойчивую, явную по времени разностную схему при аппроксимации всех пространственных производных центральными разностями. Так, схема имеет второй порядок точности по пространству и первый по времени [8].

Сложности численного моделирования переходного и турбулентного режимов течения в трубе определяются прежде всего необходимостью использования больших расчетных областей. В частности, вычислительные проблемы определяются тем, что процесс установления режимов протекает на больших длинах. Расчет нестационарного течения в трехмерной области должен проводиться без использования условий симметрии, поскольку уменьшение объема вычислительной работы за счет введения любой симметрии, в том числе симметрии начальных условий, может искажать результаты моделирования и статистическую однородность течения. Принципиальным моментом является необходимость применения разностных алгоритмов, которые не искажают физику процесса [8].

В рамках проводимого исследования также определяются языки программирования, фреймворки и библиотеки, на основе которых будет функционировать создаваемое программное средство. Эти факторы влияют на временные затраты, качество и количество издержек на разработку. Выбранный стек должен обеспечивать масштабируемость программного средства и возможность использования различными вычислительными устройствами.

Приложение планируется в качестве настольного продукта, поэтому разделение на клиентскую и серверную части не предполагается. И ПО продукта, и интерфейс будут функционировать непосредственно на вычислительном устройстве пользователя.

Программная составляющая реализуется средствами популярного языка Python, характеризующегося легкостью синтаксиса, универсальностью, поддержкой многочисленных дополнительных библиотек [9].

Во внешнем представлении библиотека PyQt включает необходимые классы для реализации привычных оконных интерфейсов с элементами управления для ввода и редактирования параметров, а также для передачи команд программе [10–12]. Библиотека разработана для программных продуктов, написанных на языке Python.

Настольные программные средства традиционно создаются с использованием СУБД SQLite, которая предусматривает табличный реляционный подход к хранению данных.

Для написания, отладки и тестирования программного кода выбрана интегрированная среда PyCharm, облегчающая разработку не только веб-приложений, но и настольных продуктов. Для большинства проектов достаточно установки бесплатной версии среды с базовым функционалом.

Описание экспериментов и их результатов

После математической постановки задачи проводится эксперимент в пакете OpenFOAM. Этапы эксперимента приведены на рисунке 1. В представленном программном средстве каждый этап осуществляется без применения терминала, через привычный оконный интерфейс.

1. Создание конфигурации расчетного случая, то есть директории, содержащей служебные папки и файлы задачи МСС. Обязательные вложенные директории – 0 (файлы с описанием

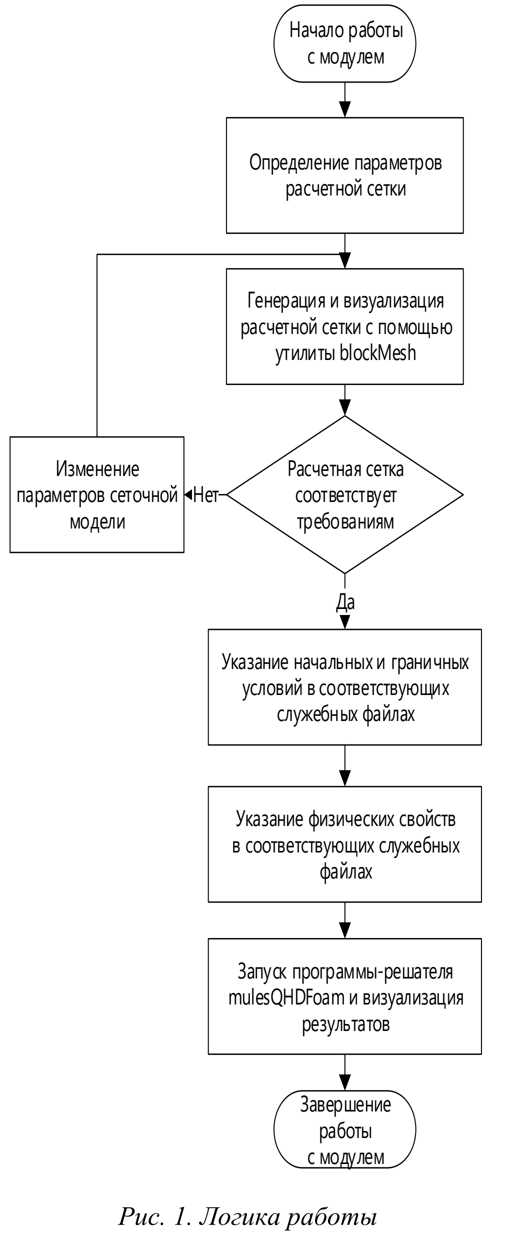

Рис. 1. Логика работы программного средства

Fig. 1. Software operation logic

начальных и граничных условий), constant (файлы с описанием геометрии задачи и свойств жидкости) и system (конфигурационные файлы с настройками решателя).

2. Построение расчетной сетки. Для решателя mulesQHDFoam достаточно построить классическую структурированную блочную сетку на базе встроенной утилиты blockMesh. Ее параметры специалист указывает в файле-словаре blockMeshDict.

3. Запуск расчета. Это старт процесса численного моделирования с помощью решателя mulesQHDFoam, завершающегося созданием численной модели с последующей ее визуализацией приложением ParaView. Данная среда требует отдельной установки, но функционирует в связке с OpenFOAM.

4. Визуализация результатов. В окне отображения ParaView специалист может анализировать изменение свойств модели в зависимости от корректировки характеристик конкретной задачи. На каждом этапе предусмотрена возможность возврата к предыдущему для изменения расчетных параметров и повторного запуска вычислений. Расчетная сетка может быть не только создана с помощью встроенных средств, но и импортирована извне.

Работа предложенного программного средства протестирована на примере задачи МСС, предусматривающей расчет течения вязкой несжимаемой изотермической жидкости. В данном случае программа-решатель mulesQHDFoam обеспечивает моделирование течений вязкой несжимаемой жидкости с тепловой конвекцией. Основными задаваемыми пользователем параметрами являются величина ускорения свободного падения (директория /constant/gravitationalProperties) и величина теплового расширения (директория /constant/thermophysical Properties). В качестве исходной жидкости рассматривается метан. Граничные условия для гидродинамических параметров указываются в файлах-словарях директории 0/. Сеточная модель генерируется с помощью утилит block Mesh и snappyHexMesh. Рисунок 2 демонстрирует основное окно графической оболочки после завершения моделирования с указанием мгновенных распределений параметра скорости в турбулентной части потока.

Обсуждение и анализ результатов

Исследование показало, что на текущий момент полноценные и полностью работоспособные графические оболочки с русифицированным интерфейсом для управления решением численных задач МСС посредством OpenFOAM отсутствуют. Использование же зарубежных программных решений либо требует существенных материальных затрат на приобретение и оплату услуг поддержки, либо, затруднено из-за англоязычного интерфейса. Кроме того, среди свободно распространяемых графических оболочек отсутствуют продукты, которые бы покрывали весь перечень задач МСС, моделируемых в OpenFOAM. Это подтверждает актуальность задачи реализации проекта отечественной оболочки для численных экспериментов на базе OpenFOAM.

Предложенное программное решение построено на многослойной архитектуре, что упрощает поддержку исходного кода и расширение функциональных возможностей. Отдельное внимание уделено визуальному оформлению интерфейса для обеспечения большей интуитивной понятности. Благодаря этому не требуется предварительная подготовка специалистов, а подробная документация упрощает работу пользователей с программным продуктом.

Многослойная архитектура проекта в перспективе позволит проводить расширение списка опций графической оболочки параллельно с внедрением в OpenFOAM новых программ-решателей и утилит пре- и постпроцессинга. Это даст возможность пользователю, не прибегая к приложениям-аналогам, выполнять численное моделирование в большинстве областей МСС.

Заключение и дальнейшие перспективы исследования

Многослойная архитектура базовой версии графической оболочки расширяет перечень возможностей приложения. Подготовлен и подключен программный модуль для управления численными экспериментами на базе решателя mulesQHDFoam платформы OpenFOAM. Таким образом, выбранный стек технологий полностью соответствует поставленным целям и позволяет решать сформулированные задачи.

Исследованы особенности применения программы-решателя mulesQHDFoam для моделирования задач МСС. Соответствующий программный алгоритм разработан с помощью выражений языка программирования Python.

За счет классов совместимой с языком Python библиотеки PyQt реализовано внешнее представление программного средства – набор

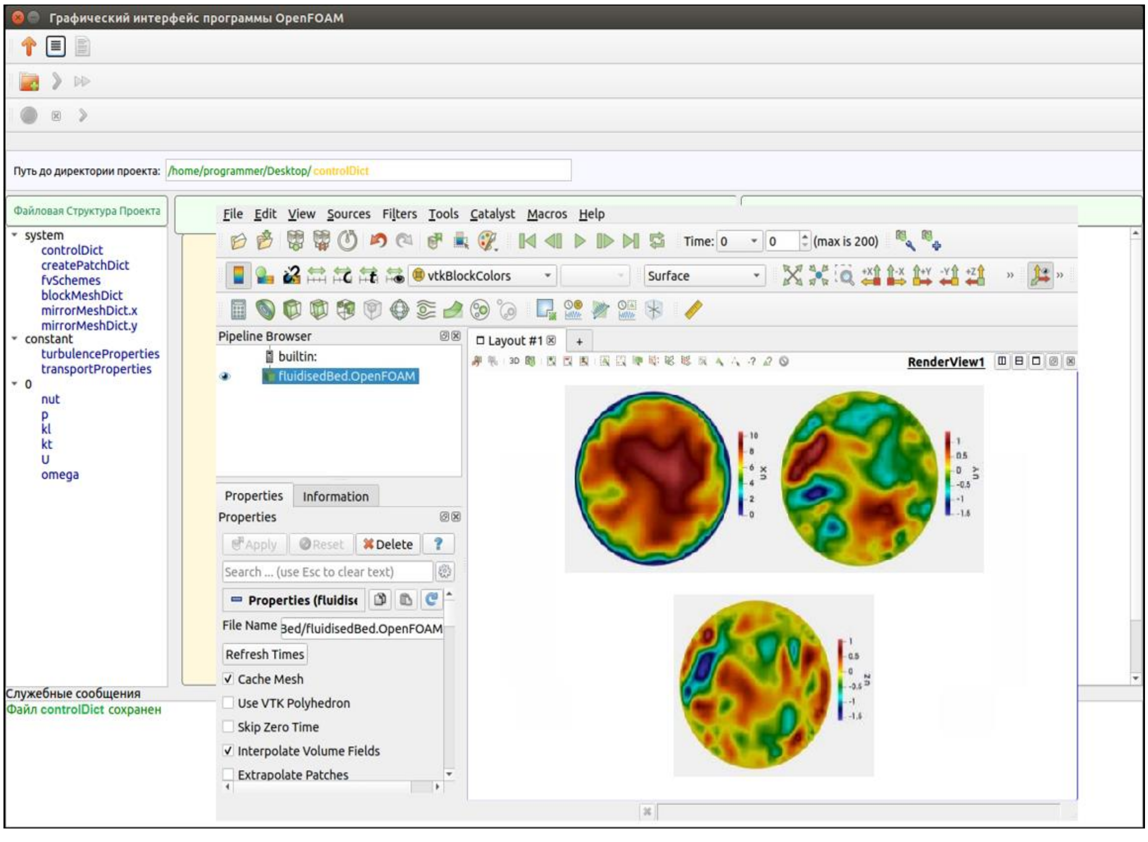

Рис. 2. Основное окно программы на этапе визуализации результатов расчетов

Fig. 2. Main program window during calculation results visualization

его экранных форм и графических элементов управления для взаимодействия с файлами-словарями расчетного случая.

Исходный код представленного программного средства, включая программную часть и внешнее представление, интегрирован в исходный код базовой версии графической оболочки. Для проекта реализовано взаимодействие с классической файловой СУБД SQLite.

Модули с исходным кодом программного средства размещены в открытом доступе (https://github.com/DmitryChitalov/OpenFOAM_GUI). Для использования продукта не требуется приобретение лицензии, допускается редактирование функционала под нужды пользователя. Проект рассчитан на применение в конструкторских и технологических подразделениях большинства предприятий отраслей тяжелого машиностроения. Благодаря графической составляющей продукта специалисты в области ракетно-космического строения отмечают снижение трудоемкости на этапах подготовки расчетного случая, запуска численного эксперимента, и постпроцессинга решения. Также наблюдаются уменьшеие количества ошибок моделирования, повышение удобства пользования.

В предложенном программном средстве реализован ряд оригинальных решений.

Так, механизм сериализации расчетных параметров со сжатием данных позволяет предварительно сохранять вводимые расчетные величины в специальных файлах для оперативного редактирования параметров. Указанные файлы сжимаются для экономии ресурсов при хранении данных.

Благодаря механизму валидации структуры расчетного случая перед запуском численного моделирования директория расчетного случая сканируется на предмет наличия всех требуемых служебных файлов, и при их отсутствии процесс моделирования не запускается. Для каждого типа задачи МСС происходит автоматическая генерация расчетного случая.

В механизме проверки типизации вводимых параметров всплывающие подсказки помогают специалисту указывать только корректные величины, при введении некорректных значений приложение не позволяет перейти на следующий шаг моделирования.

Список литературы

1. Лукин Н.В., Дзюбенко А.Л., Чечиков Ю.Б. Подходы к разработке пользовательского интерфейса // Программирование. 2020. № 5. С. 16–24. doi: 10.31857/S0132347420050052.
2. Мондрий К.П. Графический интерфейс как инструмент взаимодействия человека с компьютером // Меридиан. 2020. № 7. URL: https://meridian-journal.ru/site/article9e37/ (дата обращения 12.04.2024).
3. Пащенко Д.С. Отражение в российской практике мировых тенденций в технологиях, средствах и подходах в разработке программного обеспечения // Программная инженерия. 2017. Т. 8. № 8. С. 339–344. doi: 10.17587/prin.8.339-344.
4. Shoukat G., Siddique K., Sajid M. Modeling and CFD analyses of gear pump in OpenFOAM. Proc. ASME-JSME-KSME Joint Fluids Eng. Conf., 2019, vol. 2, art. V002T02A070. doi: 10.1115/AJKFluids2019-5435.
5. Banerjee I. Advances in model-based testing of GUI-based software. In: Advances in Computers, 2017, vol. 105, pp. 45–78. doi: 10.1016/bs.adcom.2016.12.003.
6. Читалов Д.И., Калашников С.Т. Разработка приложения для подготовки расчетных сеток с градуирующими и изогнутыми краями для программной среды OpenFOAM // Системы и средства информатики. 2018. Т. 28. № 4. С. 122–135. doi: 10.14357/08696527180412.
7. Читалов Д.И. Разработка модуля для управления моделированием системы двух сжимаемых жидких фаз с одной дисперсной фазой на базе платформы OpenFOAM // Прикладная информатика. 2024. Т. 19. № 2. С. 106–117. doi: 10.37791/2687-0649-2024-19-2-106-117.
8. Кирюшина М.А. Численный эксперимент в задаче о распространении малых возмущений в круглой трубе // Препринты ИПМ им. М.В. Келдыша. 2024. № 48. 21 с. doi: 10.20948/prepr-2024-48.
9. Сторожева В.К. Сферы применения языка Python // Актуальные проблемы авиации и космонавтики: сб. матер. VIII Междунар. науч.-практич. конф. 2022. Т. 2. С. 498–500.
10. Пальмов С.В., Поскиваткина А.А. Сравнение библиотек tkInter, PyQt, wxPython // ЕНО. 2020. № 11-2. С. 122–126.
11. Шокодько Ф.А. Разработка графического интерфейса клиентского приложения, графических элементов и их динамизация для ПК // Студент года 2023: сб. статей. 2023. С. 24–29.
12. Нехорошева Е.М., Конецкая Е.А. Графические возможности Python // ИТСиТ: матер. Всерос. научн.-практич. конф. 2021. С. 99–101.

Software & Systems doi: 10.15827/0236-235X.151.468-474 2025, 38(3), pp. 468–474

OpenFOAM with no graphical user interface: Solving the problem

Dmitry I. Chitalov [1]✉

[1] South Ural Scientific Center,
Miass, 456317, Russian Federation

For citation
Chitalov, D.I. (2025) ‘OpenFOAM with no graphical user interface: Solving the problem’, *Software & Systems*, 38(3), pp. 468–474 (in Russ.). doi: 10.15827/0236-235X.151.468-474
Article info
Received: 11.12.2024 After revision: 10.01.2025 Accepted: 11.02.2025

Abstract. Since the standard OpenFOAM distribution kit has no graphical user interface, this fact complicates user interaction with the software. This paper aims to resolve this pressing issue. It presents an integrated module enabling numerical experimentation through the mulesQHDFoam solution framework. The developed solution offers two key competitive benefits: open-source code accessibility and Russian language interface. Separation of the program code that ensures frontend functioning from the backend program code simplifies subsequent support and refinement of the module. The paper describes some aspects of using mulesQHDFoam and peculiarities in designing experiments with this solver. It also presents a list of development tools including the primary programming language, supplementary libraries, and supporting technologies. A process diagram created using the UML diagramming language demonstrates the algorithm of the user's work with the program. The author tested the module's capabilities by simulating a fundamental continuum mechanics problem. The paper presents an image of the experimental results for the selected task with a visualization of the main window of the graphical shell. Since the source code is hosted on the GitHub, third-party specialists can use the program.

The author shows the analysis of the effectiveness of the selected development technologies and determines their appropriateness for future implementation. He also presents concise results and deduces the implementation value of this research. He identifies several critical differentiators between developed modules and existing comparable software products.
Keywords: numerical modeling, OpenFOAM, mulesQHDFoam solver, open-source software, graphical user interface, Python, continuum mechanics

References

1. Lukin, N.V., Dzyubenko, A.L., Chechikov, Y.B. (2020) 'Approaches to user interface development', *Programming and Computer Software,* (5), pp. 16–24 (in Russ.). doi: 10.31857/S0132347420050052.
2. Mondry, K.P. (2020) 'Graphical user interface as a human-computer interaction tool', *Meridian Scientific Electronic J.*, (7), available at: https://meridian-journal.ru/site/article9e37/ (accessed April 12, 2024) (in Russ.).
3. Pashhenko, D.S. (2017) 'Reflection in Russian practice of world trends in technologies, tools and approaches in software development', *Software Engineering,* 8(8), pp. 339–344 (in Russ.). doi: 10.17587/prin.8.339-344.
4. Shoukat, G., Siddique, K., Sajid, M. (2019) 'Modeling and CFD analyses of gear pump in OpenFOAM', *Proc. ASME-JSME-KSME Joint Fluids Eng. Conf.,* 2, art. V002T02A070. doi: 10.1115/AJKFluids2019-5435.
5. Banerjee, I. (2017) 'Advances in model-based testing of GUI-based software', in *Advances in Computers,* 105, pp. 45–78. doi: 10.1016/bs.adcom.2016.12.003.
6. Chitalov, D.I., Kalashnikov, S.T. (2018) 'Development of an application for preparation of computational meshes with graduating and curved edges for the OpenFOAM software', *Systems and Means of Informatics,* 28(4), pp. 122–135 (in Russ.). doi: 10.14357/08696527180412.
7. Chitalov, D.I. (2024) 'Development of a module for managing the modeling of a system of two compressible liquid phases with one dispersed phase based on the OpenFOAM platform', *J. of Applied Informatics,* 19(2), pp. 106–117 (in Russ.). doi: 10.37791/2687-0649-2024-19-2-106-117.
8. Kiryushina, M.A. (2024) 'Numerical experiment in the problem of propagation of small disturbances in a round pipe', *Keldysh Institute Preprints,* (48), 21 p. (in Russ.). doi: 10.20948/prepr-2024-48.
9. Storozheva, V.K. (2022) 'Areas of application of the Python language', *Proc. VIII Int. Sci. Pract. Conf. Current Problems in Aviation and Cosmonautics*, 2, pp. 498–500 (in Russ.).
10. Palmov, S.V., Poskivatkina, A.A. (2020) 'Comparison of tkInter, PyQt, wxPython libraries', *Eurasian Sci. Association*, (11-2), pp. 122–126 (in Russ.).
11. Shokodko, F.A. (2023) 'Development of the graphical interface of the client application, graphic elements and their dynamization for a PC', *Proc. Conf. Student of the Year 2023,* pp. 24–29 (in Russ.).
12. Nehorosheva, E.M., Konetskaya, E.A. (2021) 'Graphic features of Python', *Information and Telecommunication Systems and Technologies: Proc.of the All-Russian Scientific-Practical Conf.,* pp. 99–101 (in Russ.).

Авторы
Читалов Дмитрий Иванович [1],
младший научный сотрудник, cdi9@yandex.ru

[1] Южно-Уральский научный центр,
г. Миасс, 456317, Россия

Authors
Dmitry I. Chitalov [1], Junior Researcher,
cdi9@yandex.ru

[1] South Ural Scientific Center,
Miass, 456317, Russian Federation