

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра Математического и компьютерного моделирования

Автоматизация пре-процессинга данных

в САД/САЕ системах

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

студентки 2 курса 247 группы

направление 09.04.03 — Прикладная информатика

механико-математического факультета

Симоновой Марины Рубеновны

Научный руководитель
доцент, к.т.н

И.А. Панкратов

Зав. кафедрой
зав. каф., д.ф.-м.н., доцент

Ю.А. Блинков

Саратов 2020

ВВЕДЕНИЕ

Численное моделирование в области механики сплошных сред (МСС) позволяет эффективно исследовать закономерности протекания различных физических процессов. При этом могут использоваться различные пакеты прикладных программ (CAD/CAE системы). Но не все из них обладают удобным и понятным интерфейсом пользователя, что требует от инженеров наличия специальных навыков и необходимости расходования значительных ресурсов рабочего времени на освоение интерфейсов. Одним из таких программных приложений является платформа OpenFOAM, не предусматривающая наличие GUI.

Существующий подход использования платформы OpenFOAM предусматривает использование только командной строки и требует от специалиста знания команд, предназначенных для запуска утилит OpenFOAM, необходимых для управления численными исследованиями, а также структуры каждого расчетного случая (задачи МСС) и свойств задач МСС в служебных файлах. При существующем подходе инженер вынужден работать с расчетным случаем через командную строку и вручную редактировать служебные файлы. Указанный подход весьма неудобен и требует существенных затрат рабочего времени.

Альтернативой данному подходу является разработка и использование классического оконного графического интерфейса, благодаря которому процесс численного эксперимента на основе платформы OpenFOAM будет осуществляться с применением окон и элементов управления: кнопок, выпадающих списков, текстовых полей и так далее.

Актуальность проблемы подтверждается интересом к ней зарубежных разработчиков. Ряд иностранных компаний запустили собственные проекты по созданию полноценных графических оболочек для OpenFOAM. Их результатом стали программные решения такие, как Helyx-OS, Salome и Visual-CFD, которые обеспечивают выполнение всех этапов процесса численного моделирования. Создание перечисленных GUI для взаимодействия с OpenFOAM вызвало неоднозначную реакцию среди участников международного сообщества исследователей. С одной стороны, пользователи получили возможность отказаться от трудоемкого традиционного подхода проведения численного

моделирования, а с другой – в каждой из предложенных графических оболочек были обнаружены недостатки, из-за которых не все пользователи оказались готовы к переходу от традиционного подхода к использованию оконного интерфейса. Таким образом, у исследователей сохраняется интерес к вопросу создания графической оболочки для платформы OpenFOAM.

Цель настоящей работы состоит в разработке базовой версии GUI для платформы OpenFOAM, обеспечивающей подготовку расчетного случая к решению с возможностью применения для решателя simpleFoam (стационарный несжимаемый турбулентный поток).

Данная графическая оболочка должна быть реализована в виде отдельного программного приложения и предусматривать выполнение этапа препроцессинга (определение и установка параметров задачи МСС) численного эксперимента с помощью платформы OpenFOAM.

В соответствии с обозначенной целью сформулирован следующий комплекс задач:

- 1) реализовать стартовое окно программы с возможностью выбора опции создания нового расчетного случая или выбора существующего для редактирования;
- 2) реализовать основное окно приложения с несколькими интерфейсными блоками: панелью управления, блоком подготовки исходных данных для расчетного случая, блоком отображения результатов и панелью вывода служебной информации;
- 3) в блоке подготовки исходных данных реализовать отображение окна с интерфейсными элементами управления для обеспечения возможности указания исходных параметров;
- 4) в блоке отображения результатов реализовать окно представления результатов выполнения команд;
- 5) реализовать возможность генерации моделей сеток средствами OpenFOAM (с помощью встроенных утилит blockMesh и snappyHexMesh);
- 6) реализовать возможность отображения созданной сетки с помощью инструмента постобработки ParaView, интегрированного в OpenFOAM.

1 Краткие сведения об OpenFOAM

В прикладном пакете OpenFOAM общая схема решения задач состоит из подготовки исходных данных с помощью препроцессора, собственно процесса решения, а также обработки и визуализации результатов вычислений с помощью постпроцессора. Основные задачи препроцессора - задание геометрии расчетной области, построение сетки и постановка граничных условий. В состав OpenFOAM входят препроцессоры blockMesh и snappyHexMesh. Назначением решателя является подготовка системы линейных алгебраических уравнений, аппроксимирующей исходную математическую модель, и нахождение ее решения. Все стандартные решатели в OpenFOAM поделены на группы, которые характеризуются общим подходом к решению задачи. Внутри одной группы различные решатели учитывают особенности конкретных математических моделей, а также выбор метода решения. Результатом работы решателя является набор файлов, каждый из которых содержит данные о поле исследуемой величины в конкретный момент времени. Задача постпроцессора состоит в обработке, анализе и визуализации этих данных. В одном комплекте с OpenFOAM поставляется свободно распространяемый постпроцессор ParaView.

2 Обзор графической оболочки HELYX-OS

HELYX-OS – это графический пользовательский интерфейс с открытым исходным кодом, разработанный ENGYS для работы со стандартными библиотеками OpenFOAM и предназначенный для академического использования и работы с задачами вычислительной гидродинамики базового уровня. Данный интерфейс был создан для облегчения использования стандартного OpenFOAM путем удаления длинных и сложных ручных текстовых вводов, требуемых утилитами и решателями в этом коде. Графический интерфейс обеспечивает полностью интерактивную, простую в использовании среду для выполнения всех операций пре-процессинга в процессе решения задач вычислительной гидродинамики, включая создание сетки, определение расчетного случая и выполнение решателя.

Основные особенности HELYX-OS:

- 1) Встроенная поддержка OpenFOAM и OpenFOAM+: загрузка существующих расчетных случаев путем чтения настроек непосредственно из доступных текстовых файлов проекта.
- 2) Платформы Linux и Windows (версия Windows доступна только для платных клиентов службы поддержки).
- 3) Установка одного файла: развертывание HELYX-OS и OpenFOAM в системе за один раз.
- 4) Управление утилитой snappyHexMesh, включая отображение геометрии и исполнение в графическом интерфейсе.
- 5) Комплексный модуль определения расчетного случая, включающий элементы управления для физических моделей, турбулентности, граничных условий и инициализации поля.
- 6) Вкладка процесса выполнения работы решателя с мониторингом невязки решения.
- 7) Интерфейс облачных вычислений для выполнения OpenFOAM через облачное соединение HELYX-OS, работающее параллельно.

Среди недостатков данной графической оболочки отмечены следующие: необходимость оплаты технической поддержки и отсутствие полноценной документации по продукту. Обучающая документация содержит лишь базовую инструкцию для пользователей.

3 Выбор инструментария разработки

Разрабатываемая графическая оболочка является программным средством, то есть набором файлов с программным кодом. Для его написания применяется специальный инструментарий, который включает язык программирования и среду разработки. Разрабатываемый интерфейс реализуется в виде настольного программного приложения.

Логика работы программы реализуется с помощью языка Python, который по итогам 2020 года укрепил свои позиции (согласно рейтингу TIOBE) и считается одним из наиболее популярных средств описания логики работы настольных и веб-приложений.

Для разработки графического интерфейса пользователя используется библиотека PyQt5. Данный фреймворк оптимален для реализации элементов графических интерфейсов настольных программ. Он содержит все необходимые средства для создания классических оконных интерфейсов с привлекательным дизайном.

Для хранения исходных данных расчетных случаев OpenFOAM используется NoSQL подход. Выбор указанного подхода в данной работе основывается на:

- 1) неструктурированности исходных данных OpenFOAM;
- 2) легкости масштабирования, так как объем исходных данных может нарастать в зависимости от типов экспериментов и их количества;
- 3) отсутствии необходимости в сопоставлении объекта в базе и коде информационной системы при последующей ее разработке.

В качестве системы управления базой данных выбрана MongoDB. Одними из причин такого выбора стали:

- 1) документо-ориентированное хранилище (простая и мощная JSON-подобная схема данных);
- 2) схожий с реляционными СУБД подход к хранению данных: база данных-коллекция-документ-поле;
- 3) динамический подход к данным: структура документов не определяется заранее, а отдельные коллекции и базы можно создавать прямо во время выполнения, что упрощает и ускоряет процесс разработки и позволяет динамически назначать пространства имен;

В качестве среды разработки для написания, отладки и запуска программного кода используется PyCharm. Она позволяет для каждого проекта определить нужную версию интерпретатора Python, а также в режиме реального времени установить необходимые пакеты и расширения, в том числе PyQt.

4 Проектирование системы

Разработка модели любой системы всегда предшествует ее созданию или обновлению. Это необходимо для того, чтобы яснее представить себе решаемые

мую задачу, и позволяет убедиться в архитектурной согласованности проекта до того, как он будет реализован в коде.

Метод объектно-ориентированного анализа позволяет описывать реальные сложные системы наиболее адекватным образом. Но с увеличением сложности систем возникает потребность в хорошей технологии моделирования. В качестве такой «стандартной» технологии используется унифицированный язык моделирования (UML), который является графическим языком для спецификации, визуализации, проектирования и документирования систем. С помощью UML можно разработать подробную модель создаваемой системы, отображающую не только ее концепцию, но и конкретные особенности реализации. В рамках UML-модели все представления о системе фиксируются в виде специальных графических конструкций, получивших название диаграмм.

С помощью диаграмм можно рассмотреть систему с различных точек зрения. Одна из диаграмм, например, может описывать взаимодействие пользователя с системой, другая - изменение состояний системы в процессе ее работы, третья - взаимодействие между собой элементов системы и так далее. Сложную систему можно и нужно представить в виде набора небольших и почти независимых моделей-диаграмм, причем ни одна из них не является достаточной для описания системы и получения полного представления о ней, поскольку каждая из них фокусируется на каком-то определенном аспекте функционирования системы и выражает разный уровень абстракции. Другими словами, каждая модель соответствует некоторой определенной, частной точке зрения на проектируемую систему. Набор диаграмм составляет модель системы и наиболее полно ее описывает.

При проектировании данной системы использовались диаграммы вариантов использования, активности и последовательности. Все диаграммы построены с помощью сервиса PlantText.

5 Система автоматизации пре-процессинга для OpenFOAM

Приложение OpenFOAM_PA_tool реализовано в виде набора модулей, который формирует общую структуру программы. Исходные коды програм-

мы распределены по соответствующим директориям, включенным в главную директорию OpenFOAM_PA_tool. За запуск программы отвечает файл `__main__.py`. Он содержит инструкцию отображения стартового окна приложения. Директория `./windows/` содержит служебные файлы, отвечающие за функционирование стартового, главного и диалоговых окон создания/открытия расчетного случая, а также обеспечение отображения набора форм, каждая из которых соответствует определенному блоку параметров расчетного случая. Директория `./libs/` содержит классы, отвечающие за сохранение расчетного случая в базу данных и директорию, а также загрузку расчетного случая из базы данных. Директория, `./bin/`, содержит bash-скрипты, отвечающие за запуск команд.

За инициализацию ключевых свойств сетки отвечает отдельное диалоговое окно. На основе внесенных в него значений базовых параметров автоматически генерируются параметры в остальных формах, тем самым образуя общую структуру файлов `blockMeshDict` и `snappyHexMeshDict`.

Логика функционирования форм реализуется частично в служебных файлах `bMD_window.py`, `sHMD_window.py` и `system_window.py`, в которые импортируются файлы экранных форм, и частично в самих файлах форм.

Для реализации программного запуска консольных команд OpenFOAM используется bash-скриптинг, то есть написание программы на языке shell для ее выполнения командной оболочкой. Такие программы называются сценариями, или скриптами. Автоматизированный запуск bash-сценария реализуется за счет применения модуля `subprocess` языка Python. Он обеспечивает унифицированный интерфейс доступа к операционной системе и отвечает за выполнение следующих действий: порождение новых процессов, соединение с потоками стандартного ввода, стандартного вывода, стандартного вывода сообщений об ошибках и получение кодов возврата от этих процессов. При этом сторонний процесс выносится в отдельный, параллельный главному, что позволяет пользователю осуществлять запуск любых дочерних процессов.

Таким образом, в процессе разработки программы использовались следующие подходы.

1. Создание набора форм для редактирования параметров файлов, соответствующих расчетному случаю. Например, в файлах `blockMeshDict`

и `snappyHexMeshDict` параметры сеток разделены по блокам. Каждому блоку соответствует отдельная форма набора. Заполнение форм осуществляется в строгом порядке, причем структура каждой из форм зависит от содержимого предыдущей формы. Тем самым упрощается подготовка сеточной модели пользователем.

2. Создание БД для файлов параметров расчетного случая. Для каждого из расчетных случаев создается коллекция в MongoDB, где каждый документ соответствует определенному файлу, то есть содержит структуру полей и соответствующих им значений (параметров задачи МСС). Благодаря этому механизму параметры задачи МСС могут быть не только сохранены в базе, но и загружены в соответствующую экранную форму при необходимости их редактирования.
3. Генерация `bash`-скриптов и запуск их в качестве дочерних процессов средствами языка программирования Python [16]. Использование данного подхода позволяет отказаться от привычного ручного запуска консольных команд по подготовке расчетных сеток и проведению численного моделирования на базе платформы OpenFOAM.

Основной и единственной базой данных является база `OpenFOAM_cases`. В ней находятся коллекции, названия которых соответствуют названиям расчетных случаев. Эти коллекции содержат в себе документы, каждый из которых является аналогом файла из директории расчетного случая в OpenFOAM.

В зависимости от расчетного случая документы в коллекциях имеют разную структуру, исходя из специфики решения задачи: области применения, модели сплошной среды, геометрии расчетной области, граничных и начальных условий и так далее. В таких условиях сложности структур документов документо-ориентированные БД имеют огромное преимущество по сравнению с реляционными.

ЗАКЛЮЧЕНИЕ

Традиционный подход проведения численного моделирования на базе платформы OpenFOAM требует от пользователя вручную создавать расчетные случаи, содержащие служебные директории и файлы параметров задач МСС, а также использовать командную строку для управления ходом численного моделирования. Он также требует знания структуры служебных файлов и названий параметров задачи из этих файлов. Такой подход не обеспечивает экономию временных затрат и усложняет освоение платформы OpenFOAM. Поэтому в качестве альтернативы могут использоваться специализированные графические оболочки.

В рамках данной работы реализована базовая версия графической оболочки для подготовки исходных данных численного эксперимента на базе платформы OpenFOAM с помощью языка программирования Python 3.5 и фреймворка PyQt5. Программа реализована в виде десктопного приложения с графическим интерфейсом, включающим в себя стартовое окно, основное окно с панелью управления, блоком подготовки исходных данных для расчетного случая, блоком отображения результатов и панелью вывода служебной информации. Данное приложение предоставляет возможность:

- 1) подготовки нового расчетного случая и открытия существующего для внесения изменений или проверки параметров. В рамках данной опции возможна подготовка расчетных случаев с несложной геометрией (когда внутри домена расположен простой объект, например, шар, пластина или конус) и указанием начальных условий для скорости и давления, а также базовых параметров решателя simpleFoam;
- 2) генерации моделей сеток средствами OpenFOAM (с помощью встроенных утилит blockMesh и snappyHexMesh);
- 3) запуска и остановки процесса решения, запущенного посредством решателя simpleFoam. В процессе решения осуществляется подсчет давления и скорости при ламинарном режиме течения;
- 4) визуализации результатов моделирования с помощью инструмента постобработки ParaView, интегрированного в OpenFOAM.

Программный запуск консольных команд OpenFOAM реализуется посредством вызова bash-скриптов с помощью модуля subprocess языка Python.