Тема диплома: “Разработка инструмента моделирования и визуализации формы пространственной кривой, восстановленной по данным, изменяющимся в реальном времени”.

В современном мире одним из наиболее важных направлений развития технологий является создание новых технических устройств, способных улучшать качество и эффективность процессов в различных сферах деятельности человека. Одним из таких устройств является гибкая протяженная гидроакустическая буксируемая антенна (ГПБА) с датчиками системы ориентации, которая может использоваться в многих областях, таких как морское исследование, геологические исследования, метеорология и другие.

Однако одной из проблем, с которой сталкиваются исследователи, работающие с данными, получаемыми от таких устройств, является сложность восстановления формы гибкой антенны по этим данным. Традиционно, обработка гидроакустической информации ГПБА выполняется в предположении о прямолинейной форме ГПБА. С включением датчиков системы ориентации в современные ГПБА появилась возможность получать информацию о текущей форме антенны в реальном времени и учитывать её в дальнейшей обработке. Датчики системы ориентации позволяют получить свои координаты в пространстве, но в практических случаях этих датчиков значительно меньше, чем акустических. Задача восстановления формы антенны состоит в получении координат каждого акустического приёмника и её можно сформулировать как интерполяцию кривой в пространстве.

Для решения этой проблемы в данной работе предлагается реализовать инструмент моделирования и визуализации формы пространственной кривой, восстановленной по данным, изменяющимся в реальном времени.

Для успешного достижения поставленной цели необходимо решить следующие задачи:

1. Провести анализ существующих методов восстановления
2. Реализовать инструмент визуализации
3. Произвести тестирование разработанного продукта

Для анализа существующих методов восстановления нужно определится с критериями, которые будут оказывать наибольшее влияние на программный продукт. Среди множества критериев выделяются следующие:

1. Сложность алгоритма – характеристика, которая указывает на то, сколько времени, либо какой объем памяти потребуется для выполнения.
2. Наличие готовой реализации позволит ускорить разработку продукта, а также снизить количество ошибок и некачественного кода.
3. Гибкость настройки отражает способность метода настраиваться под конкретную задачу, что благоприятно сказывается на точности вычисления, но увеличивает количество аспектов, на которые стоит обратить внимание при разработке

Опираясь на проведённый анализ существующих алгоритмов восстановления данных, можно сформировать требования, которых должна придерживаться разрабатываемая программная система:

1. Продукт должен включать один или несколько алгоритмов восстановления.
2. В программной системе должны использоваться методы, реализованные в готовых математических библиотеках.
3. Программная система должна иметь возможность менять условия или параметры используемых алгоритмов интерполяции в зависимости от ситуации и предоставлять удобный пользовательский интерфейс.

Таким образом разрабатываемая система, производящая моделирование и визуализацию пространственной кривой по восстановленным данным, удовлетворит всем требованиям. Это позволит пользователям в реальном времени отслеживать ориентацию гибкой буксируемой антенны в пространстве во время буксировки.

При проектировании и разработке приложения был использован архитектурный паттерн MVC (Model-View-Controller). Этот паттерн позволил разделить приложение на три компонента: модель, представление и контроллер. Каждый компонент выполняет свою задачу, что облегчает сопровождение и развитие приложения. На слайде представлена обобщенная архитектура решения.

Уровень контроллера в приложении отвечает за обработку пользовательского ввода и управление взаимодействием между компонентами приложения. Он является посредником между пользователем и моделью, а также управляет представлением, которое отображает данные, полученные от модели.

В данном случае, контроллер обрабатывает различные пользовательские действия, такие как добавление, удаление или изменение точек кривой, выбор определенного вида кривой для визуализации, изменение масштаба и поворота камеры в 3D пространстве и т.д. Контроллер также должен обеспечить правильное взаимодействие с моделью, так что любые изменения, сделанные пользователем, отображены в представлении.

Уровень представления отвечает за отображение данных пользователю и обработку пользовательского ввода. В данном случае, главная страница приложения включает в себя два основных элемента:

Панель настройки, которая задает базовую модель, известные точки и метод восстановления, а также управляет визуальной составляющей отображающегося графика.

Поле визуализации, которое отображает график, построенный на основе введенных пользователем параметров. В этом поле пользователь может видеть исходную модель, известные точки, а также восстановленную кривую и взаимодействовать с ней (вращать, приближать/отдалять). На уровне представления также реализована валидация пользовательского ввода.

Уровень модели отвечает за хранение данных и бизнес-логику приложения. Он является независимым от пользовательского интерфейса и контроллера.

Модель представляет собой совокупность информации об истинной форме модели, наборе известных точек, а также восстановленной форме. Кроме того, модель включает блок обработки отклонений и производительности, который производит замеры времени и отвечает за сравнение восстановленной формы с исходной моделью.

Восстановленная форма включает в себя доступ к математическим библиотекам для выполнения процесса интерполяции и построения аппроксимирующих функций на основе имеющихся данных.

Основную часть всего главного окна занимает поле визуализации смоделированного графика. В данном окне пользователь имеет возможность выбрать наиболее подходящий для себя угол обзора, соответствующее приближение и в общих чертах оценить отклонение выбранного метода восстановления от базовой модели.

Меню настроек предоставляет богатый функционал по настройке исходных данных. Пользователь может влиять на следующие аспекты:

* Длину истинной модели
* Функцию истинной модели
* Количество, цвет и размер всех отображаемых точек
* Наличие отклонений и их максимальные значения по всем осям
* Тип метода восстановления
* Отображаемую информацию при выборе конкретной точки
* Интервал с которым будет обновляться эмуляция
* Шаг на который по графику будет продвигаться эмуляция
* Осуществлять запуск или остановку эмуляции

Тестирование GUI приложения включает в себя проверку правильности отображения графических элементов интерфейса, работоспособности кнопок и других элементов управления, а также общей удобности и эргономики пользовательского интерфейса.

Оценка интерфейса производится вручную по методике “белого ящика”, которая подразумевает доступ к исходному коду приложения что поможет проводить более глубокий анализ его работы.

В ходе выполнения тестирования были разобраны все элементы управления, масштабируемость, реакция системы на взаимодействия с ними, а также указаны допустимые значения валидации.

Во время выполнения работы был произведен обзор существующих методов восстановления. Определены требования для успешного написания программного продукта. Произведено тестирование графического интерфейса.

В дальнейшем приложение может быть расширено путем добавления других алгоритмов восстановления в виде подключаемых библиотек.