

Виртуальный полигон для задач обеспечения безопасности судоходства

Седова Нелли Алексеевна

*Морской государственный университет им. адм. Г.И. Невельского
доцент*

Седов Виктор Александрович

*Морской государственный университет им. адм. Г.И. Невельского
заведующий кафедрой*

Аннотация

В работе предлагается один из подходов к созданию виртуального полигона в приложении к задачам обеспечения безопасности судоходства. Кратко описаны основные подсистемы разрабатываемого виртуального полигона для моделирования различных ситуаций, возникающих в море: подсистема визуализации, физический движок, графический пользовательский интерфейс, а также подсистема звукового сопровождения.

Ключевые слова: виртуальный полигон, моделирование, безопасность мореплавания

The virtual polygon for tasks of ensure the safety of navigation

Sedova Nelly Alekseevna

*Maritime State University named after G.I. Nevelskoy
assistant professor*

Sedov Viktor Aleksandrovich

*Maritime State University named after G.I. Nevelskoy
head of department*

Abstract

The paper proposes an approach to the creation of the virtual polygon for tasks of ensure the safety of navigation. Briefly described the main subsystems of the virtual polygon developed for modeling various situations arising at sea: subsystem of visualization, physics engine, graphical user interface and the audio subsystem.

Keywords: virtual polygon, modeling, safety of navigation

Разработка и внедрение методов и средств обеспечения безопасного плавания в современных условиях судоходства предваряется экспериментальными исследованиями, проведение которых затруднительно в связи с необходимыми значительными материальными затратами. В связи с этим разработка так называемых виртуальных полигонов, т.е. программно-аппаратных средств, позволяющих исследователям моделировать различные

ситуации, возникающие в море, в том числе ситуации, предшествующие столкновениям морских судов с разными объектами, встречаемыми на маршруте, представляется крайне актуальной. Целью настоящей работы является представление одной из возможных технологий создания виртуального полигона применительно к задачам обеспечения безопасности плавания (ВПОБП). Достижению поставленной цели предшествовало решение следующего комплекса задач.

- Проведение поиска по нахождению имеющихся в открытом доступе моделей судов, а также наземных и находящихся в море объектов.
- Создание трёхмерных моделей судов (на примере танкера и малого рыболовного бота) для исследования систем определения степени опасности столкновения [1, 2] и систем расхождения судов [3, 4].
- Определение возможных ситуаций, моделирующих ветро-волновые возмущения, а также поиск средств получения физической интерпретации воздействия внешних погодных факторов на модели судов и др. объектов.
- Разработка системы визуализации морских поверхностей, составление диаграммы классов системы визуализации.
- Разработка графического пользовательского интерфейса (ГПИ) виртуального полигона.
- Разработка и отладка программного комплекса виртуального полигона для моделирования различных задач обеспечения безопасности судовождения.

Научная значимость представленных результатов обусловлена созданием моделей (скорости ветра и волнения, физического их воздействия на наземные и находящиеся в море объекты), адекватных реальным. Практическая значимость заключается в том, что результаты проведённой работы представляют собой комплекс программ для моделирования и визуализации морских судов, возможных внешних климатических воздействий на них, морских поверхностей, а также различных объектов, которые потенциально могут находиться на маршруте движения судна.

В развитие теоретических и практических основ разработки виртуальных полигонов внесли вклад авторы следующих исследований. В статье [5] авторы представили проект виртуального полигона для исследования динамики движения судов, включающего банк данных моделей судов, базу знаний ветро-волнового поля и динамики судов, блок миссий и навигаций и ряд др. блоков. В исследовании [6] представлены результаты исследований, направленных на изучения взаимодействия разноуровневых приложений виртуального полигона, моделирующих поведение морских объектов. Авторы работ [7-10] представляют виртуальный полигон для задач исследовательского проектирования морских судов и средств освоения океана, находящихся под воздействием нерегулярных внешних возмущений. В работах [11-13] представлены результаты по разработке многофункционального вычислительного комплекса для исследования

динамики взаимодействия объектов, имеющих различную физическую природу.

При разработке ВПОБП использовались следующие программные средства: для написания системы визуализации использовался язык C++, для вывода трёхмерной графики – возможности библиотеки Open Graphic Library (OpenGL) версии 4.3., для симуляции физических процессов – открытый физический движок Bullet Physics Library (BPL). Заметим также, что для создания платформенно-независимого приложения использован компилятор GCC, входящий в ядро Linux.

Фрагментарно опишем технологию разработки следующих подсистем виртуального полигона для моделирования различных ситуаций, возникающие в море: подсистему визуализации, физический движок, графический пользовательский интерфейс, а также подсистему звукового сопровождения.

Система визуализации состоит из двух частей: препроцессора и рендерера. Препроцессор служит для подготовки к рендерингу, т.е. процессу получения изображения по модели, в пространстве каждой из заданных камер. Рендерер отрисовывает предобработанные объекты. Препроцессор подготавливает матрицы для рендера и собирает в список все объекты, видимые текущей камерой. Видимые объекты сортируются в порядке приближения к камере.

Система визуализации ВПОБП является управляющим классом-оболочкой над препроцессором и рендерером, через который осуществляется управление параметрами рендеринга. Здесь же производится регистрация трёхмерных моделей, регистрация шейдеров, управление параметрами шейдеров, назначение текстур.

Для создания окна приложения и инициализации контекста OpenGL вызывается функция `SetDimension()`, при этом никакие другие операции, имеющие отношение к OpenGL, например, загрузка шейдеров, текстур или моделей невозможна до вызова функции `SetDimension()`.

Для работы с текстурами использовалась структура-оболочка `Texture_descriptor`, содержащая в себе имя файла, идентификатор и тип текстуры. Текстуры могут быть загружены, назначены и удалены с помощью менеджера ресурсов в любое время. Текстурные дескрипторы могут использоваться для загрузки текстуры в шейдер.

Загрузка составных частей шейдерной программы осуществляется через менеджер ресурсов, а их линковка производится в системе визуализации функцией `SetUpShader()`.

Для задания входных параметров шейдерной программы использовалась параметризованная функция `DefineShaderInputParam()`, имеющая две перегруженных версии – для задания константных значений и `Callback` функций. `Callback` функции используются для задания вычисляемых, часто меняющихся значений входных параметров.

Неотъемлемой частью загрузки полигона является регистрация трёхмерных моделей в OpenGL, которая производится генерацией всех

используемых моделей вершинного буфера. Эту операцию выполняет функция `SetUpModels()`. Если, например, при переходе в другую локацию, требуется загрузить в память несколько новых моделей, то это потребует полного пересоздания вершинного буфера, а удаление старого произойдёт автоматически. В работе используются трехмерные модели следующих объектов:

- морские суда (танкера и сухогрузы), рыболовные суда, яхты;
- низколетящие аппараты (квадрокоптеры, вертолеты, самолеты небольших габаритов);
- отдельные плавучие объекты – небольшие объекты, находящиеся наплаву и свободно дрейфующие, такие как мусор, бочки, контейнера и т.п.;
- стационарные навигационные объекты – буи, радиобуи, радиолокационные маяки и т.п.;
- элементы ландшафта – острова, полуострова, мысы, осыхаемые участки береговой черты и др.;
- метеорологические явления – облака нижнего слоя, сильные дожди и др. осадки;
- человек за бортом.

Препроцессор считывает из очереди сообщения, содержащие информацию для рендеринга каждого кадра, и выполняет следующие действия:

- производит матричные преобразования;
- рассчитывает расстояния от камеры до всех объектов сцены, проверяет, какие объекты попадают в сферу видимости, и сортирует их в порядке удаления от камеры;
- отбрасывает объекты, которые не попали в усеченную пирамиду видимости;
- результат предобработки передается на рендерер.

Рендерер перед использованием инициализируется, далее функция `RenderCameraFrame()` получает пакет настроек для кадров, соответствующих каждой камере, и отсортированный список моделей в области видимости, и для каждого трёхмерного объекта вызывает функцию `RenderObject()`. Завершается рендерер кадра вызовом функции `RenderFinalFrame()`, которая отображает все камеры и графический пользовательский интерфейс на экран. Пример рендеринга показан на рисунке 1.

Мир в пространстве каждой камеры рендерится в текстуру, которая позже может быть использована для непосредственного отображения на весь экран или на его часть с использованием привязки к элементу ГПИ.



Рисунок 1 – Результат рендеринга сцены

Физический движок, используемый в настоящем ВПОБП, основан на физическом движке реального времени BPL, который распространяется под свободной лицензией zlib. Скомпилированная библиотека и исходный код движка BPL размещены в свободном доступе на хостинге Google Code. Библиотека BPL позволяет производить расчеты столкновений различных фигур (сфера, параллелепипед, цилиндр, конус, выпуклый корпус и сетка треугольников), поддерживает физику объемных тел, а также гидродинамику сглаженных частиц (жидкости, газы), при этом реализованы два основных алгоритма расчета столкновений: алгоритм Гилберта – Джонсона – Кёрти и алгоритм определения минимальной глубины проникновения двух выпуклых тел [14].

Все объекты, участвующие в расчетах физических взаимодействий, обладают следующими параметрами: сила гравитации, масса, центр массы, скорость, габаритный контейнер. Движок BPL производит аппроксимацию объекта геометрическими примитивами, а также полностью контролирует перемещение объектов в пространстве, для чего содержит структуру WorldTransform, хранящую позиции всех физических объектов сцены и методы для получения или изменения позиций отдельных объектов.

Обнаружение столкновений, используемое в BPL, имеет широкую и узкую фазы: во время широкой фазы алгоритм обнаружения столкновений

уменьшает число потенциально взаимодействующих объектов, во время узкой – происходит генерация точек контакта между парами соприкоснувшихся объектов.

Основной интерфейс анализа текущего положения объектов (т.е. обновление структуры WorldTransform) – StepSimulation, который вызывает функцию VtMotionState (метод анализа объектов сцены), которая обновляет позиции объектов, игнорирует неподвижные объекты для оптимизации вычислений. Функция VtMotionState также вызывается при создании объекта, запоминая его изначальное состояние, она также содержит методы, которые устанавливают или получают позицию отдельного объекта.

Графический пользовательский интерфейс состоит из нескольких основных частей: классы контролов, звено дерева, итератор для дерева и менеджер ГПИ. Классы контролов реализуют функции всех компонентов, они имеют общего родителя (базовый класс, от которого происходит наследование базовых функций). Звено дерева представляет собой звено дерева для итератора, который хранит в себе один контрол и все его свойства. Итератор для дерева связывает все звенья между собой в виде дерева, где имеются свои уровни и система вложений. Менеджер ГПИ занимается начальной инициализацией графического пользовательского интерфейса, через него происходит все управление ГПИ.

При начальной инициализации ГПИ происходит загрузка всех заранее подготовленных изображений внешнего вида контролов и их размеров, их расположениями и номерами шаблонов. После инициализации через менеджер ГПИ можно программно добавлять новые контролы или удалять имеющиеся, а также осуществлять поиск контролов по имени, получать или устанавливать для них параметры.

Подсистема звукового сопровождения реализована с использованием кроссплатформенной свободно распространяемой библиотеки Open Audio Library (OpenAL) и её высокоуровневой модификации – ALUT, предоставляющей API для работы с аудиоданными в формате .wav. Встроенный в библиотеку рендерер производит необходимые вычисления для определения расстояния от источника звука до слушателя и расчета эффекта Доплера. Для работы со звуком был создан класс, использующий функции высокого уровня библиотеки ALUT, для загрузки, воспроизведения аудиоданных, а также для добавления различных эффектов.

В ходе проведенной работы создана основа для разработки виртуального полигона для исследования систем обеспечения безопасности плавания (ВПОБП версии 1.0), а именно частично или полностью решены следующие задачи.

- Осуществлен поиск с целью нахождения имеющихся в открытом доступе моделей судов, а также наземных и находящихся в море объектов. Такой поиск необходимо проводить эпизодически для добавления новых объектов в ВПОБП.
- Созданы трёхмерные модели танкера и малого рыболовного бота.

- Определены ветро-волновые возмущения, определен физический движок VPL для расчетов столкновений различных объектов.
- Разработана система визуализации морских поверхностей, требующая дополнительного наполнения новыми поверхностями.
- Разработан графический пользовательский интерфейс ВПОБП.
- Разработана версия 1.0 ВПОБП, произведена её отладка.

Дальнейшие исследования будут посвящены созданию новых объектов для наполнения ВПОБП, моделированию внешних погодных воздействия на модели судов и др. объектов, а также улучшению системы визуализации.

Библиографический список

1. Sedov V.A., Sedova N.A., Glushkov S.V. The fuzzy model of ships collision risk rating in a heavy traffic zone // Vibroengineering PROCEDIA. 2016. Т. 8. С. 453-458.
2. Sedova N.A., Sedov V.A., Glushkov S.V. The fuzzy model of the emergency level assessment at sea // Vibroengineering PROCEDIA. 2016. Т.8. С. 506-511.
3. Sedova N.A. A logical-linguistic model of ship collision avoidance in a heavy traffic zone // В сборнике: 21st Saint Petersburg International Conference on Integrated Navigation Systems, ICINS 2014 - Proceedings 21. 2014. С. 166–170.
4. Седов В.А., Седова Н.А. Гибридная система управления судном // Научно-технический вестник Поволжья. 2013. № 2. С. 204-207.
5. Вунна Чжо, Дегтярев А.Б. Виртуальный полигон по исследованию динамики судов // Интернет-журнал Науковедение. 2014. №6 (25). URL: <http://cyberleninka.ru/article/n/virtualnyy-poligon-po-issledovaniyu-dinamiki-sudov> (дата обращения: 13.12.2016).
6. Соз Моэ Лвин. Разработка элементов виртуального полигона моделирования окружающей морской среды в гетерогенном вычислительном окружении: дис. канд. техн. наук: 05.13.18. Санкт-Петербург, 2011. 235 с.
7. Безгодков А.А. Виртуальный полигон для исследования морских объектов в экстремальных условиях эксплуатации: автореф. дис. канд. техн. наук: 05.13.18. Санкт-Петербург, 2011. 18 с.
8. Безгодков А.А., Бухановский А.В. Виртуальный полигон для исследования экстремальной динамики морских объектов на нерегулярном волнении // Известия вузов. Приборостроение. 2011. №05. С. 98-100.
9. Безгодков А.А., Загарских А.С., Бухановский А.В. Визуализация динамики морских объектов в широкоэкранных системах виртуальной реальности // Научно-технический вестник информационных технологий, механики и оптики. 2011. № 3 (73). С. 84-88.
10. Безгодков А.А., Иванов С.В., Косухин С.С., Бухановский А.В. Виртуальный полигон для исследовательского проектирования морских объектов и сооружений. Известия высших учебных заведений.

- Приборостроение. 2011. Т. 54. № 10. С. 58-64.
11. Бугакова Н. Ю., Волкогон В. А., Нечаев Ю. И. Виртуальная среда современной теории катастроф при контроле чрезвычайных ситуаций в морской индустрии // Морские интеллектуальные технологии. 2014. № 2 (24). С. 24-29.
 12. Бухановский А.В., Васильев В.Н., Нечаев Ю.И. Виртуальный полигон – центр компетенции на основе современной теории катастроф // Материалы 3-й Всероссийской научно-технической конференции: в 2 т. – Ростов-на-Дону. Издательство Южного федерального университета, 2014. С. 191-196.
 13. Бухановский А.В., Иванов С.В., Ковальчук С.В., Нечаев Ю.И. Программный комплекс «Виртуальный полигон – центр компетенции», обеспечивающий оперативный контроль чрезвычайных ситуаций в транспортной и социальной среде на основе облачных вычислений и ГРИД-технологий // Суперкомпьютерные технологии (СКТ-2014) // Материалы 3-й Всероссийской научно-технической конференции: в 2 т. – Ростов-на-Дону. Издательство Южного федерального университета, 2014. С. 197-202.
 14. Ермолин Е. Н. Методы определения и разрешения столкновений на полигональных моделях: дис. канд. физико-математических наук: 05.13.18. Новосибирск, 2011. 155 с.