

Рисунок 6а – Вероятности результирующих состояний надёжности (S2–S5)

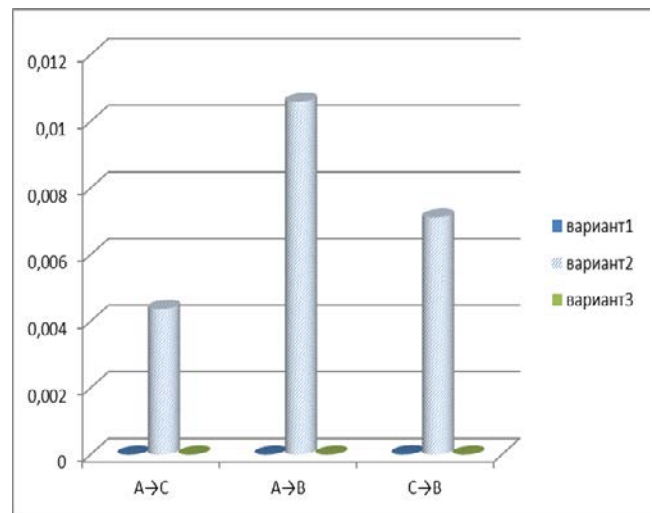


Рисунок 6б – Вероятности надёжного соединения терминальных вершин системы (рисунок 4)

Анализируя данные двух вариантов организации электроэнергетической системы, можно сделать вывод о выигрышном варианте 1 по сравнению с вариантом 2, что наглядно видно на рисунках 5б и 6б, поскольку увеличение вероятностей соединения терминальных вершин свидетельствует о частом срабатывании выключателей при аварийном состоянии оборудования выделенных линий.

Представленный в статье подход эффективен при проектировании новых электроэнергетических объектов, при подключении новых линий электропередачи, оценке характеристик надёжности трансформаторных подстанций с целью повышения надёжности и безопасности организации всей системы. Его применение позволяет анализировать изменение надёжности организации электроэнергетических объектов в результате изменения параметров надёжности их элементов.

Литература

1 Сукач, Е. И. Моделирование вероятностных характеристик сложных систем с использованием стохастических алгебр / Е. И. Сукач и др. // V Международная конференция-форум «Информационные системы и технологии», Академия управления при Президенте Республики Беларусь, 16–17 ноября 2009 г. – Минск : А. Н. Вараксин – 2009. – ч. 1. – С.178–181.

2 Жердецкий, Ю. В. Способ формализации объектов графовой структуры с вероятностными параметрами функционирования / Е. И. Сукач, Д. В. Ратобылская, Ю. В. Жердецкий, Г. А. Мальцева // Известия Гомельского государственного университета им. Ф. Скорины. – 2012. – № 5(74). – С. 195–202.

ИСПОЛЬЗОВАНИЕ ФИЗИЧЕСКИХ ЗАКОНОВ ДЛЯ ОПИСАНИЯ ВЗАИМОДЕЙСТВИЙ ТРЕХМЕРНЫХ ФИЗИЧЕСКИХ ОБЪЕКТОВ

В статье спроектирована и сконструирована программная библиотека расширений, которая инкапсулирует в себе весь сложный математический код и позволяет пользователю оперировать объектами трехмерного мира. Библиотека содержит методы, оперирующие математическими объектами, обрабатывает их взаимодействие и позволяет достаточно быстро создавать графические приложения средней сложности, используя готовые шаблоны, с возможностью подключения физических взаимодействий – соударения, изменения динамических характеристик, различные соединения нескольких объектов в один, гравитационные взаимодействия. Для работы с библиотекой используется язык программирования C# и платформа для разработки .NET Framework.

В современной промышленности широко используются различные виды моделирования. В первую очередь это моделирование различных производственных объектов. Часто также важно создать модель, составляющие которой будут изменять свои динамические характеристики, взаимодействовать с другими объектами.

Трехмерное моделирование и анимация стали весьма популярными в сфере архитектуры. Многие чертежные схемы до сих пор выполняются с помощью карандаша или программ двухмерного моделирования, однако все больше проектов реализуется посредством трехмерной графики. Полученные изображения или анимация используются в качестве демонстрационного материала для показа клиентам, инвесторам и плановым отделам. В некоторых случаях трехмерные модели комбинируют с фотографиями фактической местности, таким образом, получая чрезвычайно реалистичную визуализацию здания на запланированном участке [1].

Трехмерное моделирование очень популярно в компаниях, занимающихся промышленным дизайном, поскольку эта технология позволяет представить полноценную модель изделий на экране, а затем по желанию клиента изменить, например, ее цвет, стиль или какой-либо элемент. Соответствующее программное обеспечение также часто применяется автомобильными дизайнерами для проектирования корпусов, поскольку эти специалисты имеют возможность оценить сложное взаимодействие света и тени на контурах транспортного средства.

Объемное моделирование (solid modeling) представляет собой разновидность трехмерного моделирования, которая применяется в инженерных приложениях. В подобных приложениях используется информация о весе материала, его плотности, пределе прочности и другие реальные факты, добавляемые в набор данных (dataset) модели. Поскольку вся физическая информация о «реальных» материалах становится доступной для использования, проектируемые мосты, корпуса кораблей и детали машин можно подвергнуть имитируемому на компьютере давлению. В результате, инженеры получают возможность проверить, соответствует ли проектируемый объект поставленным условиям, для чего не придется создавать его физический прототип.

В инженерной сфере интенсивно используются программы CAD/CAM (computer-aided design – автоматизированное проектирование/computer-aided manufacturing – автоматизированное производство), с помощью которых дизайнерские чертежи можно запрограммировать непосредственно в оборудовании, производящем детали. Кроме того, такая технология макетного моделирования, как лазерная стереолитография, позволяет быстро воссоздать физические прототипы сложных объектов, отливая их из жидкой пластмассы уровень за уровнем. В технологии FDM (Fused Deposition Modeling)

тонкая пластмассовая нить, намотанная на катушку, подается в нагретый наконечник с помощью компьютера и формируется по тонким уровням для формирования целой детали. В данный момент ведутся дискуссии относительно установки системы FDM на международной космической станции для создания простых сменных деталей прямо на рабочем месте. Для моделирования различных объектов удобно использовать графические приложения. Они предоставляют большое количество готовых к использованию шаблонов, а также возможности их настройки. Однако не всегда такие программные пакеты предлагают описать физические взаимодействия между графическими объектами.

Представление данных на мониторе компьютера в графическом виде впервые было реализовано в середине 50-х годов для больших ЭВМ, применявшихся в научных и военных исследованиях. С тех пор графический способ отображения данных стал неотъемлемой принадлежностью подавляющего числа компьютерных систем, в особенности персональных. Графический интерфейс пользователя сегодня является стандартом «де-факто» для программного обеспечения разных классов, начиная с операционных систем. Существует специальная область информатики, изучающая методы и средства создания и обработки изображений с помощью программно-аппаратных вычислительных комплексов, – компьютерная графика. Она охватывает все виды и формы представления изображений, доступных для восприятия человеком либо на экране монитора, либо в виде копии на внешнем носителе (бумага, киноплёнка, ткань и прочее). Без компьютерной графики невозможно представить себе не только компьютерный, но и обычный, вполне материальный мир. Визуализация данных находит применение в самых разных сферах человеческой деятельности. Для примера назовем медицину (компьютерная томография), научные исследования (визуализация строения вещества, векторных полей и других данных), моделирование тканей и одежды, опытно-конструкторские разработки. В зависимости от способа формирования изображений компьютерную графику принято подразделять на растровую, векторную и фрактальную.

Отдельным предметом считается трехмерная графика, изучающая приемы и методы построения объемных моделей объектов в виртуальном пространстве. Как правило, в ней сочетаются векторный и растровый способы формирования изображений. Особенности цветового охвата характеризуют такие понятия, как черно-белая и цветная графика. На специализацию в отдельных областях указывают названия некоторых разделов: инженерная графика, научная графика, Web-графика, компьютерная полиграфия и прочие. На стыке компьютерных, телевизионных и кинотехнологий зародилась и стремительно развивается сравнительно новая область компьютерной графики и анимации. Заметное место в компьютерной графике отведено развлечениям. Появилось даже такое понятие, как механизм графического представления данных (Graphics Engine). Рынок игровых программ имеет оборот в десятки миллиардов долларов и часто инициализирует очередной этап совершенствования графики и анимации. Хотя компьютерная графика служит всего лишь инструментом, ее структура и методы основаны на передовых достижениях фундаментальных и прикладных наук: математики, физики, химии, биологии, статистики, программирования и множества других. Это замечание справедливо как для программных, так и для аппаратных средств создания и обработки изображений на компьютере. Поэтому компьютерная графика является одной из наиболее бурно развивающихся отраслей информатики и во многих случаях выступает «локомотивом», тянущим за собой всю компьютерную индустрию.

Для создания трехмерной графики используются специальные программы, которые называются редакторы трехмерной графики, или 3D-редакторы. 3ds max 7 является одной из таких программ. Результатом работы в любом редакторе трехмерной графики, в том числе и в 3ds max 7, является анимационный ролик или статическое изображение, просчитанное программой. Чтобы получить изображение трехмерного объекта, необходимо создать в программе его объемную модель.

В ходе работы был сконструирован набор программных библиотек, позволяющих создавать и отображать графические трехмерные объекты. Также, на основе пакета Farseeer Physics Engine, разработана модель взаимодействия графических объектов – соударения, изменения динамических характеристик, различные соединения нескольких объектов в один, гравитационные взаимодействия и так далее.

В разработанной программной библиотеке были использованы так называемые графические примитивы, на основе которых выстраивались более сложные структуры. Примитивы показаны на рисунке 1.

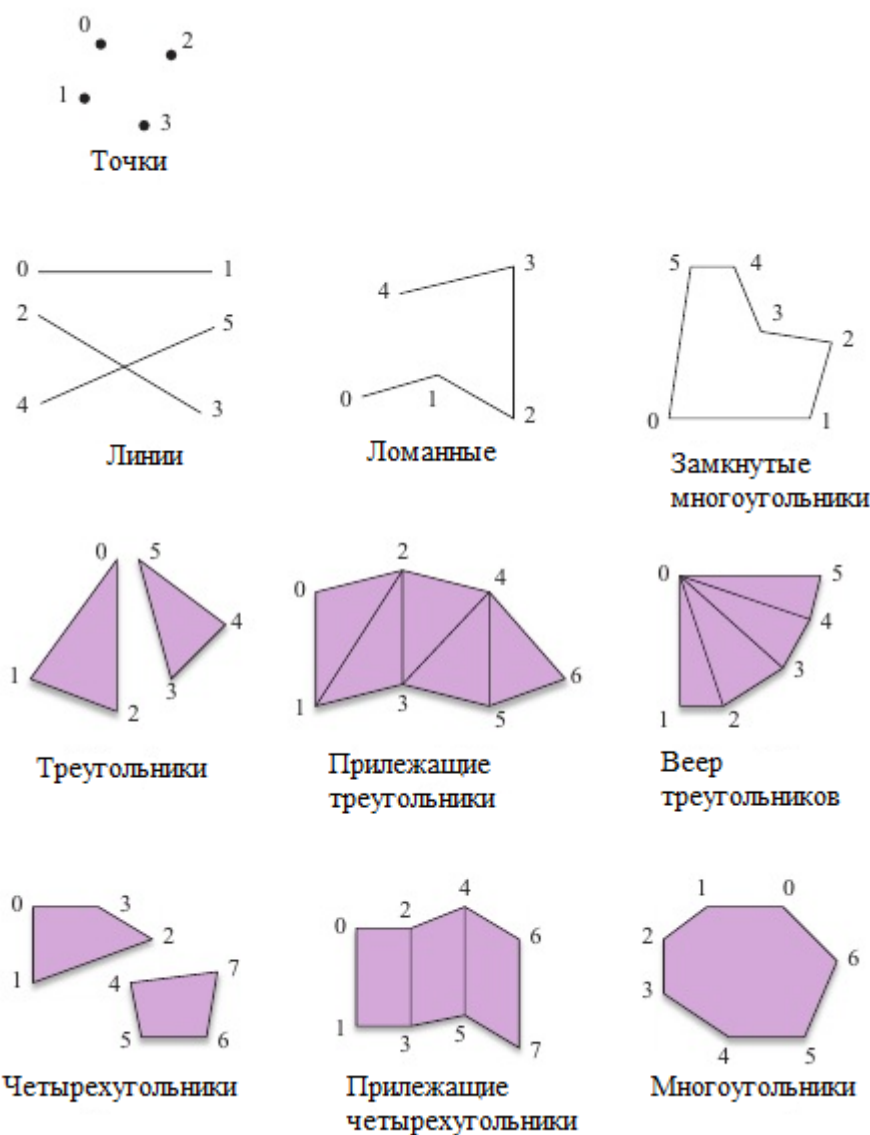


Рисунок 1– Графические примитивы

Разработанная библиотека позволяет конструировать различные графические модели, используя готовые шаблоны, с возможностью подключения физических взаимодействий. Для работы с библиотекой используется язык программирования C# и платформа для разработки .NET Framework [2]. Далее представлен исходный код инициализации приложения, использующего библиотеку.

```
namespace Demo1
{
    public partial class Page : UserControl
```

```

{
    private Storyboard gameLoop;
    private PhysicsSimulator physicsSimulator;
    private List<PhysicsBox> entityList = new List<PhysicsBox>();
    public Page()
    {
        InitializeComponent();
        gameLoop = new Storyboard();
        gameLoop.Completed += new EventHandler(gameLoop_Completed);
        gameLoop.Duration = TimeSpan.FromMilliseconds(25); //The game will be updated
every 25 millisecond
        this.Resources.Add(«gameLoop», gameLoop);
        this.Loaded += new RoutedEventHandler(Page_Loaded);
        physicsSimulator = new PhysicsSimulator(new Vector2(0, 150));
        PhysicsBox box = new PhysicsBox(128, 128, 100);
        box.Body.Position = new Vector2((float)this.Width / 2, (float)this.Height / 2);
        physicsSimulator.Add(box.Body);
        entityList.Add(box);
        this.DrawingCanvas.Children.Add(box);
    }
    protected void Page_Loaded(object sender, RoutedEventArgs e)
    {
        gameLoop.Begin();
    }
    protected void gameLoop_Completed(object sender, EventArgs e)
    {
        physicsSimulator.Update(.01f); ИСПОЛЬЗОВАНИЕ ФИЗИЧЕСКИХ ЗАКОНОВ
        ДЛЯ ОПИСАНИЯ ВЗАИМОДЕЙСТВИЙ ТРЕХМЕРНЫХ
        ФИЗИЧЕСКИХ ОБЪЕКТОВ
        foreach (PhysicsBox box in entityList)
        {
            box.Update();
        }
        gameLoop.Begin();
    }
}
}

```

Литература

1 Samuel R. Buss. 3D Computer Graphics, Cambridge / Samuel R. Buss / Cambridge. – 2003. – С. – 304–350.

2 Cawood S. Microsoft XNA Game Studio Creators Guide: An Introduction to XNA Game Programming / Cawood S. / McGraw-Hill Osborne Media, 2007. – 322 с.

УДК 517.444

И. С. Ковалева