

ПРИМЕНЕНИЕ АРХИТЕКТУРЫ OPENGL ДЛЯ ПРЕДСТАВЛЕНИЯ РЕЗУЛЬТАТОВ РАСЧЕТОВ В СИСТЕМАХ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ

М. В. Литвинов (Санкт-Петербург)

В статье рассматривается применение и преимущества архитектуры OpenGL в программах имитационного математического моделирования. На примере программы, описывающей процесс полного пуска с неподвижного носителя одного снаряженного боекомплекта РСЗО «СМЕРЧ» [1]. Изучение данного процесса функционирования связано с существенными изменениями параметров математической модели, описывающей последовательный сход с направляющих пусковой установки N реактивных снарядов, общая масса которых соизмерима с массой самого носителя.

Решение задачи динамики старта заключается в определении сил и моментов, действующих на пусковую установку при старте реактивных снарядов с направляющих. Это позволит повысить точность стрельбы путем компенсирования воздействий, оказываемых на пусковую установку сходящими реактивными снарядами. Наибольшую актуальность решение задачи динамики приобрело применительно к реактивным системам залпового огня последнего поколения, таким как РСЗО «СМЕРЧ» (рис. 1).



Рис. 1

Причина этого – большая масса снарядов, соизмеримая с массой самой пусковой установки, и вследствие этого, возникающие при старте большие моменты. По имеющимся данным, масса снаряда составляет 800 кг, в то время как масса пустого пакета направляющих – 2000 кг. Данные получены из открытых источников, размещенных в сети Internet [2].

Динамика старта описывается системой уравнений Лагранжа переменного порядка. Система уравнений Лагранжа в матричной форме выглядит так:

$$\ddot{q} = -D^{-1} (H + C - \tau) \quad (1)$$

где \ddot{q} – вектор обобщенных ускорений компонентов системы;

D – матрица перехода;

H – вектор внутренних сил в системе;

C – вектор внешних сил, действующих на систему;

τ – вектор обобщенных сил.

Старт одной ракеты распадается на три промежутка времени: промежуток времени набора ракетой тяги в 20 тонн, промежуток времени движения ракеты по направляющим и промежуток времени успокоения колебаний пусковой установки после схода

ракеты. На промежутках времени схода ракеты с направляющих порядок системы равен трем, т. к. система включает в себя вращающуюся часть, качающуюся часть и реактивный снаряд. Уравнение Лагранжа для трехзвеньевого системы запишется так:

$$\ddot{q} = -D^{-1}(H + C - \tau) = -D^{-1} \begin{pmatrix} \dot{q}^T H_1 \dot{q} \\ \dot{q}^T H_2 \dot{q} \\ \dot{q}^T H_3 \dot{q} \end{pmatrix} - D^{-1}C + D^{-1}\tau \quad (2)$$

На остальных промежутках времени порядок системы равен двум, т. к. реактивный снаряд выпадает из системы. Уравнение Лагранжа для двухзвеньевого системы запишется так:

$$\ddot{q} = -D^{-1}(H + C - \tau) = -D^{-1} \begin{pmatrix} \dot{q}^T H_1 \dot{q} \\ \dot{q}^T H_2 \dot{q} \end{pmatrix} - D^{-1}C + D^{-1}\tau \quad (3)$$

Для решения системы уравнений Лагранжа изначально предложено использовать математический пакет Matlab®, однако в ходе работы выяснилось, что время, затрачиваемое пакетом на расчет одного залпа, оказывается слишком велико. За счет создания для решения задачи динамики старта специальной программы скорость расчета была увеличена на порядок, что позволило говорить о моделировании залпа в реальном времени на компьютере уровня Pentium IV. Программа была написана на языке C++ в среде Borland C++ 5.01 и оптимизирована под данную задачу.

Интерфейс пользователя создан в среде визуального программирования Borland Builder 5.0. В программе применяются совместно классы VCL и библиотека OpenGL 1.2. При таком сочетании используемых инструментов, программист получает возможность с меньшими затратами времени создать интерфейс приложения и включить в него демонстрационные вставки, написанные на основе библиотеки OpenGL, которая поставляется вместе с компилятором.

Библиотека OpenGL представляет собой программный интерфейс для аппаратного обеспечения машинной графики. Этот интерфейс состоит приблизительно из 250 отдельных команд, которые используются для того, чтобы определить объекты и операции, необходимые для создания интерактивных прикладных программ. Для достижения универсальности в состав библиотеки OpenGL не включены никакие команды для выполнения задач работы с окнами или для пользовательского ввода. Поэтому все управление программой осуществляется при помощи стандартных классов VCL.

Библиотека OpenGL не представляет команды высокого уровня для описания моделей трехмерных объектов. Такие команды могли бы позволить определить относительно сложные формы, например автомобиль, части тела, самолеты и т. д.

При использовании библиотеки OpenGL модели создаются из ограниченного набора геометрических примитивов – точек, линий и многоугольников. Спецификой представления этих примитивов заключается в возможности рисования, как на плоско-

сти, так и в определенном объеме. Каждая точка определяется тремя координатами X, Y, Z. В частном случае можно задать глубину Z равной нулю. Тогда все построения будут происходить на плоскости.

В при работе с интерактивными приложениями большую роль играет механизм выбора объектов. В ходе работы объекты могут изменять свое положение в пространстве: перемещаться и поворачиваться вокруг любой оси, что может затруднить определение объекта, который пользователь выбрал в трехмерной сцене. Для решения данной проблемы библиотека OpenGL обеспечивает механизм выбора, который автоматически сообщает, какие объекты написаны внутри указанной области окна. Можно использовать этот механизм вместе со специальной сервисной программой для определения того, какие объекты находятся в пределах области, заданной пользователем, или выбраны курсором.

В режиме выбора информация о рисунке возвращается приложению вместо того, чтобы направиться в буфер кадра, как это делается в режиме исполнения. Таким образом, экран остается неизменным, пока OpenGL находится в режиме выбора.

Основное преимущество архитектуры OpenGL в использовании графического процессора видеокарты, что позволяет без замедления расчетов представить более наглядные результаты моделирования. К тому же данная библиотека поддерживается большинством операционных систем мира, что делает код написанный с ее использованием, универсальным и переносимым.

В программе моделирования пуска пакета из 12 ракет с неподвижного основания интерфейс взаимодействия программы с пользователем разделен на две части. Информация о параметрах пуска вводится при помощи стандартных функций GDI (время между стартом очередных ракет, очередность старта, полное время пуска и координаты поворота пакета относительно неподвижной платформы) (рис. 2). Все данные, вводимые в исходных данных проверяется на достоверность.

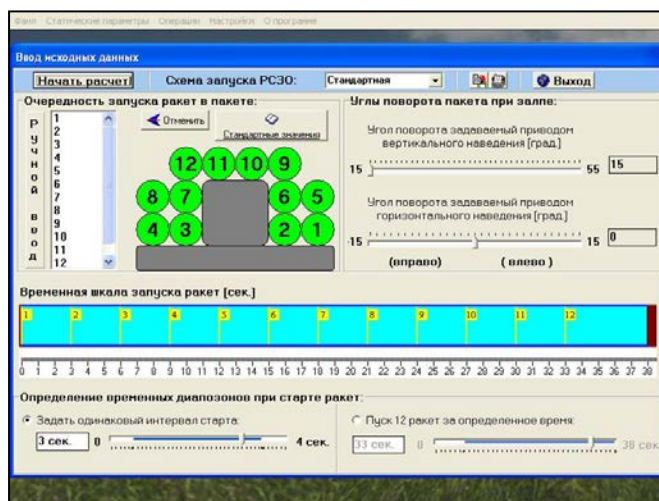


Рис. 2

Вторая часть реализована с использованием библиотеки OpenGL. В программе рисуется модель РСЗО «Смерч» по внешнему виду, такая же как настоящая (рис. 3). Во время старта очередной ракеты пользователь может наблюдать колебания всего пакета в соответствии с расчетными значениями в реальном или в псевдо-реальном времени (замедленный режим). Так же при нажатии курсором мыши на интересующие пользователя детали модели он получает весь набор предусмотренных графиков, характеризующих этот узел.

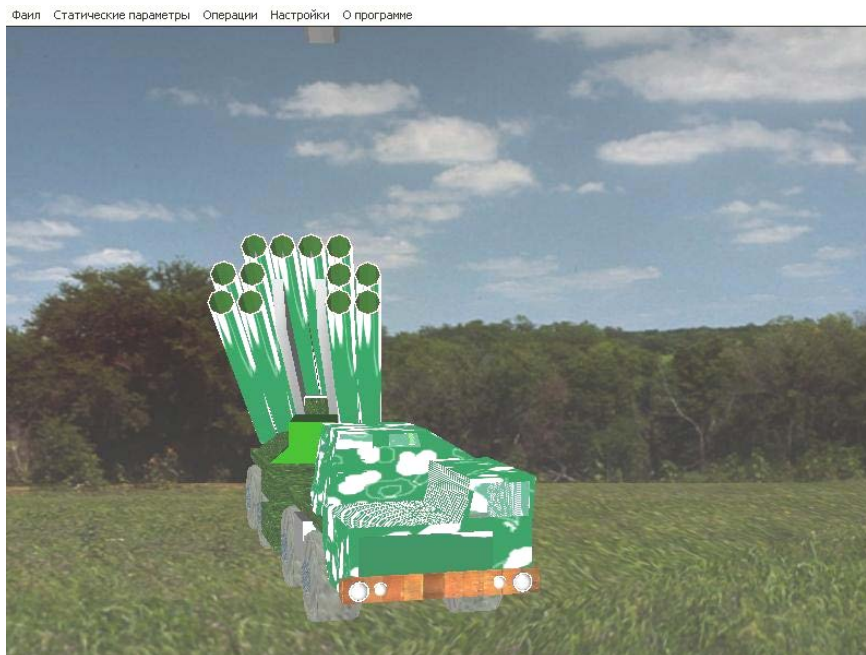


Рис. 3

Подведем некоторые итоги. Использование 3D модели установки позволяет наглядно представить процессы, проходящие в реальной модели, которые в реальной жизни не поддаются наблюдению. В процессе моделирования были получены результаты о колебании пакета направляющих комплекса на 5–10 градусов от линии прицеливания во время залпа, но самое главное, пакет не успевает стабилизироваться за трехсекундный промежуток времени между пусками. А это значит, что для дальнейшего увеличения точности прицеливания комплекса необходимо стабилизировать пакет направляющих. Для этого есть два пути: введением стабилизирующих приводов управляемых компьютерной системой, либо при помощи модели определить оптимальное сочетание очередности пуска и времени между очередными пусками. Для обоих путей применима данная программа. Таким образом, можно глубже провести анализ применяемых математических моделей и облегчается процесс выявления возможных ошибок, а также повысить наглядность, удобство, скорость работы программы, использовав для этого аппаратные возможности современных компьютеров при помощи OpenGL.

Литература

1. Визуализация функционирования стартовой установки за полный пуск боекомплекта одного снаряженного пакета.– Отчет о НИР № Г12 – 449. – ЗАО «РИВАС». – Санкт-Петербург. 2003.
2. «Кирилл и Мефодий» – «Интернет». – 2002: Система «Смерч». – www.km.ru.
3. **Тарасов И.А.** Основы программирования OpenGL.-М.: Горячая линия – телеком, 2000.- 187 с.: ил.
4. **Тихомиров Ю.В.** Программирование трехмерной графики в Visual C++. – СПб.: ВHV – СПб, 1998. – 256 с.: ил.
5. **Томпсон Н.** Секреты программирование трехмерной графики для Windows 95: Пер. с англ. -СПб.: Питер, 1997. – 352 с., ил.
6. **Тюкачев Н, Свиридов Ю.** Delphi 5 создание мультимедийных приложений.-СПб.: ПИТЕР, 2001.-398 с., ил.